

零基础轻松学会自动化技术丛书

零基础轻松学会

西门子S7-200PLC

李长军 主编

关开芹 李长城 副主编



机械工业出版社
CHINA MACHINE PRESS

VISIT…

LANZAROTE
Caliente.com

零基础轻松学会自动化技术丛书

零基础轻松学会西门子 S7-200 PLC

李长军　主　编
关开芹　李长城　副主编



机 械 工 业 出 版 社

本书是“零基础轻松学会自动化技术丛书”之一。本书共分7章，以西门子S7-200系列PLC为例介绍。主要内容包括PLC的基础知识；S7-200 PLC基本指令；步进顺序控制；S7-200 PLC功能指令；PLC网络通信技术应用；常用扩展模块；常用机床电气的PLC改造实例。附录中给出了S7-200 PLC指令集、特殊存储器（SM）标志位及错误代码。

本书的编写注重实用性，突出应用能力的提高，起点低，内容结构完整，条理清晰，语言通俗，趣味性强，图文结合，易学易懂，结构安排符合认知规律。

本书适合作为从事自动化应用的电气技术人员自学或培训教材，也可作为大中专院校、技校及职业院校电气专业的教材和参考书使用。

图书在版编目（CIP）数据

零基础轻松学会西门子S7-200 PLC/李长军主编. —北京：机械工业出版社，2014.7

（零基础轻松学会自动化技术丛书）

ISBN 978-7-111-47481-4

Ⅰ.①零… Ⅱ.①李… Ⅲ.①plc技术 Ⅳ.①TM571.6

中国版本图书馆CIP数据核字（2014）第169934号

机械工业出版社（北京市百万庄大街22号 邮政编码100037）
策划编辑：徐明煜 责任编辑：徐明煜
版式设计：赵颖喆 责任校对：肖 琳
封面设计：路恩中 责任印制：刘 岚
北京圣夫亚美印刷有限公司印刷

2014年10月第1版第1次印刷
184mm×260mm·17.25印张·417千字
0001—3000册
标准书号：ISBN 978-7-111-47481-4
定价：49.90元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务	网络服务
社服务中心：(010)88361066	教材网：http://www.cmpedu.com
销售一部：(010)68326294	机工官网：http://www.cmpbook.com
销售二部：(010)88379649	机工官博：http://weibo.com/cmp1952
读者购书热线：(010)88379203	**封面无防伪标均为盗版**

前　　言

随着科技的迅速发展，生产生活中的电气自动化程度越来越高，越来越多的人正在或者将要从事自动控制工作。而 PLC 实现的工业控制应用尤为普遍，为了让大家能跟上新技术发展，迅速掌握 PLC 技术，我们特编写了《零基础轻松学会西门子 S7-200 PLC》一书。

在本书的编写过程中，我们主要贯彻了以下编写原则：

1. 根据职业岗位需求入手，精选教材内容。本书以西门子 S7-200 PLC 为例，主要介绍了 PLC 的基本知识、基本指令、步进顺控指令、功能指令、通信控制和模拟量控制等，并在此基础上，深入浅出地介绍了相关的经典控制程序。

2. 本书突出以“图表”来说明问题。书中通过用不同形式的图片和表格，让读者轻松、快速、直观地学习 PLC 的有关知识，尽快适应电气工作岗位的需求。

3. 本书突出以技能为主，以能力为本位，淡化理论，强化实用性。书中较好地处理了理论与实践技能的关系，在“理论够用”的基础上，突出应用性和职业性的特点，注重对分析实际问题、解决实际问题能力的培养。

本书突出职业技术教育特色，可作为初、中、高等电气技术人员指导用书和中等职业学校、高职院校电类专业参考用书。

本书由李长军任主编，关开芹、李长城任副主编，沈东辉、卢强、咸晓燕、李宗金、郭庆玲、卢旭辰、肖云也参加了编写工作。

在编写中，由于作者水平有限，书中错误在所难免，恳切希望广大读者对本书提出宝贵的意见和建议，并请发送到邮箱 lydgxh@ 163. com，以便今后加以修改完善。

目　录

前言

第一章　PLC 的基础知识 …… 1
　第一节　PLC 的组成与工作原理 …… 1
　　一、PLC 的外形 …… 1
　　二、PLC 的基本结构 …… 5
　　三、PLC 的工作原理 …… 8
　　四、PLC 的特点与分类 …… 9
　第二节　S7-200 PLC 的编程元件及语言 …… 10
　　一、基本数据类型与寻址方式 …… 10
　　二、PLC 的编程元件 …… 13
　　三、PLC 的编程语言 …… 14
　第三节　STEP7-Micro/WIN V4.0 编程软件的安装与操作 …… 16
　　一、系统安装要求 …… 17
　　二、软件安装步骤 …… 17
　　三、认识 STEP7-Micro/WIN V4.0 编程软件主界面 …… 19
　　四、计算机与 PLC 的通信连接 …… 23
　　五、一个简单程序的编辑与调试运行 …… 26
　第四节　PLC 常用外部设备与接线 …… 32
　　一、PLC 的输入设备与接线 …… 32
　　二、PLC 的输出设备与接线 …… 36

第二章　S7-200 PLC 基本指令 …… 39
　第一节　位逻辑指令 …… 39
　　一、指令介绍 …… 39
　　二、指令练习 …… 40
　　三、梯形图的编程规则 …… 41
　第二节　定时器与计数器 …… 43
　　一、指令介绍 …… 43
　　二、指令练习 …… 44
　第三节　PLC 基本指令应用实例 …… 47
　　实例 1　PLC 控制三相异步电动机连续运行 …… 47

实例 2　PLC 控制三相异步电动机正反转运行 …… 49
实例 3　PLC 控制三相异步电动机Y-△减压起动 …… 52
实例 4　PLC 控制三台电动机顺序起停 …… 54
实例 5　电动机位置自动往返控制 …… 55
实例 6　长定时的 PLC 控制 …… 58
实例 7　闪光灯的 PLC 控制 …… 59
实例 8　单按钮 PLC 控制电动机起停 …… 60
第三章　步进顺序控制 …… 62
第一节　顺序控制及顺序功能图 …… 62
一、顺序控制概述 …… 62
二、顺序功能图 …… 62
三、步进顺控指令 …… 65
第二节　单流程结构步进顺序控制 …… 66
一、单流程结构顺序功能图 …… 66
二、单流程结构的编程 …… 66
第三节　选择结构步进顺序控制 …… 67
一、选择结构顺序功能图 …… 67
二、选择结构的编程 …… 67
第四节　并行结构步进顺序控制 …… 69
一、并行结构顺序功能图 …… 69
二、并行结构的编程 …… 69
第五节　步进顺序控制的综合应用实例 …… 70
实例 1　简易红绿灯控制系统 …… 70
实例 2　两种液体混合装置控制系统 …… 71
实例 3　简易洗车控制系统 …… 75
实例 4　机械臂大小球分选系统 …… 80
实例 5　十字路口交通灯控制系统 …… 81
实例 6　三层电梯的 PLC 控制 …… 86
第四章　S7-200 PLC 功能指令 …… 107
第一节　数据传送指令 …… 107
一、指令介绍 …… 107
二、指令练习 …… 109
第二节　比较操作指令 …… 110
一、指令介绍 …… 110
二、指令练习 …… 113
第三节　循环指令 …… 115
一、指令介绍 …… 115
二、指令练习 …… 116

第四节　移位指令 …… 116
一、指令介绍 …… 116
二、指令练习 …… 119
第五节　四则运算指令 …… 122
一、指令介绍 …… 122
二、指令练习 …… 125
第六节　跳转指令 …… 126
一、指令介绍 …… 126
二、指令练习 …… 127
第七节　转换指令 …… 127
一、指令介绍 …… 128
二、指令练习 …… 130
第八节　子程序与中断指令 …… 133
一、指令介绍 …… 133
二、指令练习 …… 138
第九节　逻辑运算指令 …… 140
一、指令介绍 …… 140
二、指令练习 …… 142
第十节　高速计数器指令 …… 143
一、指令介绍 …… 143
二、指令练习 …… 151
第十一节　高速脉冲输出指令 …… 153
一、指令介绍 …… 153
二、指令练习 …… 160
第十二节　功能指令综合应用实例 …… 166
实例 1　彩灯的闪烁控制 …… 166
实例 2　四路抢答器控制 …… 167
实例 3　花样喷泉控制 …… 168
实例 4　PLC 与步进电动机的运动控制 …… 174

第五章　PLC 网络通信技术应用 …… 181
实例 1　两台 S7-200 PLC 的 PPI 通信 …… 181
实例 2　S7-200 PLC 指令向导编程的 PPI 通信 …… 186
实例 3　两台 S7-200 PLC 的 Modbus 通信 …… 191
实例 4　S7-200 PLC 的以太网通信 …… 197

第六章　常用扩展模块 …… 209
第一节　扩展模块介绍 …… 209
一、S7-200 CPU 数字量扩展模块 …… 209
二、S7-200 主机的模拟量扩展模块 …… 209

三、特殊功能模块 …… 212
第二节 扩展模块的应用 …… 212
一、I/O 点数扩展和编址 …… 212
二、模拟量扩展模块的应用 …… 213

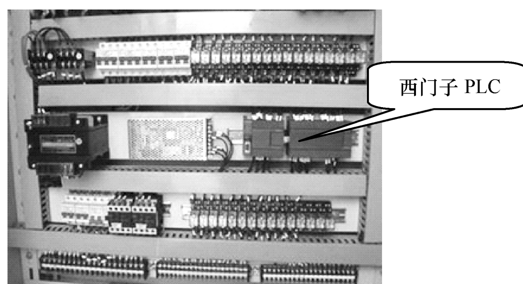
第七章 常用机床电气的 PLC 改造实例 …… 220
实例 1 CA6140 卧式车床的 PLC 控制 …… 220
实例 2 X62W 万能铣床的 PLC 控制 …… 225
实例 3 Z3040 摇臂钻床的 PLC 控制 …… 234
实例 4 M7120 平面磨床的 PLC 控制 …… 240

附录 …… 247
附录 A S7-200 的 SIMATIC 指令集 …… 247
附录 B S7-200 的特殊存储器（SM）标志位 …… 252
附录 C S7-200 的错误代码 …… 264

第一章

PLC 的基础知识

1969 年美国数字设备公司（DEC）研制出了世界上第一台 PLC，并在美国通用汽车（GM）公司的汽车生产线上首次应用成功，实现了工业生产的自动化。随着电子技术和计算机技术的发展，PLC 也在不断完善中。近年来，PLC 集电控、电仪、电传为一体，性能更加优越，已成为自动化工程的核心设备，如右图所示。



PLC 在工业生产设备中的应用图

1987 年，国际电工委员会（IEC）颁布了可编程序控制器（PLC）标准草案第三稿，在草案中对 PLC 进行了如下定义：

可编程序控制器（PLC）是一种数字运算操作的电子系统，专为在工业环境下应用而设计。它采用可编程序的存储器，用来在其内部存储执行逻辑运算、顺序控制、定时、计数和算术运算等操作的指令，并通过数字的、模拟的输入和输出，控制各种类型的机械或生产过程。可编程序控制器及其有关设备，都应按易于与工业控制系统形成一个整体，易于扩充其功能的原则设计。

第一节　PLC 的组成与工作原理

一、PLC 的外形

德国西门子公司是世界上生产 PLC 的主要厂商之一，其生产的电子产品以性能精良而久负盛名，其产品涵盖了微型、小型、中型和大型等各种类型的 PLC。目前主要流行SIMATIC S7-200/200CN、SIMATIC S7-300/300C 和 SIMATIC S7-400 三大系列 PLC 产品，如图 1. 1-1所示。西门子公司的产品不断推陈出新，1995 年底推出的 S7 系列 PLC 产品取代



a) S7-200 系列 PLC　b) S7-300系列 PLC　c) S7-400系列 PLC

图 1. 1-1　西门子公司的可编程序控制器

了原来的 S5 系列 PLC，1998 年 S7-200 PLC 升级为第二代产品，2004 年又升级为第三代产品，在 2005 年底又推出了面向中国市场的 SIMATIC S7-200CN 系列产品。

图 1. 1-2 为 S7-200 CPU226 模块实物图。下面介绍西门子 S7-200 CPU226 模块。

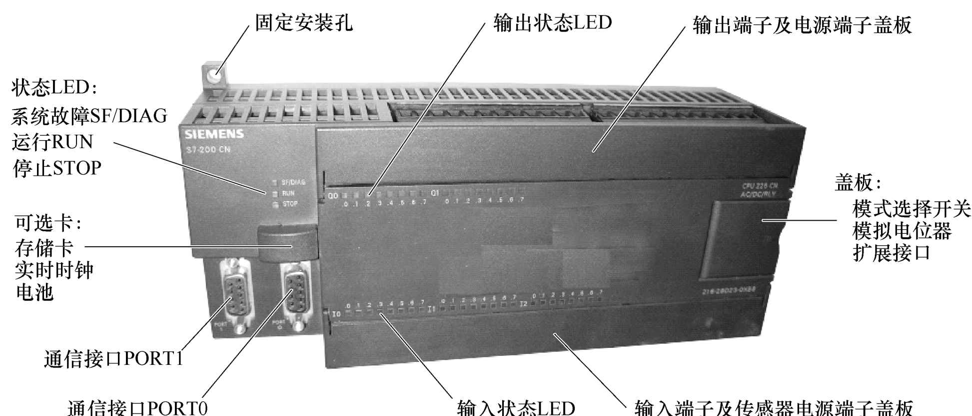


图 1. 1-2　CPU226 模块实物

1. CPU 模块的型号

CPU 模块的型号如图 1. 1-3 所示。

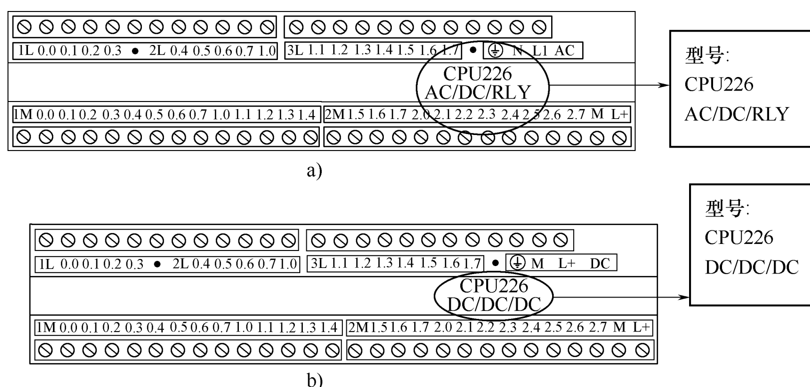


图 1. 1-3　CPU 模块的型号

【型号解读】

每一种型号的 CPU 模块都有 DC24V 和 AC120～220V 两种电源供电的类型。例如，CPU226 有 CPU226 DC/DC/DC 和 CPU226 AC/DC/RLY 两种。图 1. 1-3 中 AC/DC/RLY 的含义：AC 表示供电电源电压为交流 220V，DC 表示输入端的电源电压为直流 24V，RLY 表示继电器输出；DC/DC/DC 说明是 24V 直流电源供电、直流数字量输入、晶体管直流数字量输出。

2. 输入与输出（I/O）接线端子

在 CPU 模块的面板底部、顶部都有一排接线端子。底部一排接线端子是输入信号的接入端子及传感器电源端子。顶部一排接线端子是输出信号的输出端子及 PLC 的供电电源端子。

图 1.1-4 为 CPU226 模块的 I/O 及电源接线端子示意图。

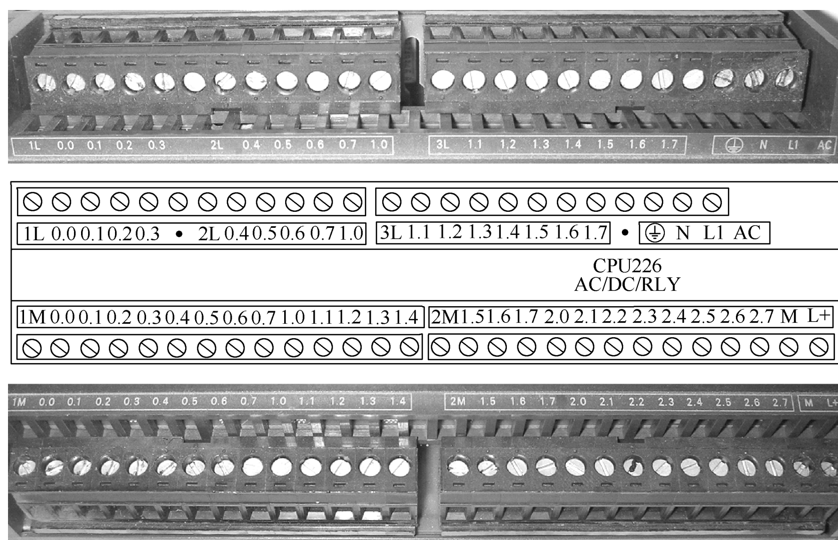


图 1.1-4 CPU226 模块的电源及 I/O 接线端子

CPU226 模块 I/O 端子共 40 点，24 个输入点（I0.0 ~ I0.7、I1.0 ~ I1.7 及 I2.0 ~ I2.7）和 16 个输出点（Q0.0 ~ Q0.7 和 Q1.0 ~ Q1.7）。在编写端子代码时采用八进制，没有 0.8、0.9、1.8、1.9 等。



解 读

1. 输入端子

1）I0.0 ~ I1.4：第一组输入继电器端子。

2）I1.5 ~ I2.7：第二组输入继电器端子。

3）1M、2M：第一、二组输入继电器的公共端口。

2. 传感器电源接线

1）M：内部 DC 24V 电源负极，接外部传感器负极或输入继电器公共端。

2）L +：内部 DC 24V 电源正极，为外部传感器或输入继电器供电。

3. 输出端子

1）Q0.0 ~ Q0.3：第一组输出继电器端子。

2）Q0.4 ~ Q1.0：第二组输出继电器端子。

3）Q1.1 ~ Q1.7：第三组输出继电器端子。

4）1L、2L、3L：第一、二、三组输出继电器的公共端口。各组输出之间是互相独立的，这样负载可以使用多个电压系列（如 AC 220V、DC 24V 等）。

5）●：带黑点的端子上不要外接导线，以免损坏 PLC。

4. PLC 电源接线

⏚：接地线；N：中线；L1：电源相线，交流电压为 85 ~ 265V。

3. I/O 状态指示灯与运行状态指示灯

1）在 CPU 模块的面板下方和上方分别有一排状态指示灯（LED），分别指示输入和输出的逻辑状态。当输入或输出为高电平时，LED 亮，否则不亮。

2）在 CPU 模块的左侧有三个运行状态指示灯（LED），分别指示系统故障/诊断（SF/DIAG）状态、运行（RUN）状态和停止（STOP）状态。

4. S7-200 CPU 的工作模式

S7-200 CPU 的工作模式有停止（STOP）模式和运行（RUN）模式两种，要改变工作模式有以下三种方法。

1）使用 CPU 模块上的模式开关。揭开 CPU 模块的前盖，模式开关有三个转换位置：RUN、TERM（终端）和 STOP。开关拨到 RUN 时，CPU 模块运行程序，即 PLC 按照扫描周期循环执行用户程序，但此时不能向 PLC 写入程序；开关拨到 STOP 时，CPU 模块停止运行程序，即 PLC 停止执行用户程序，但此时可以利用编程设备向 PLC 写入程序，也可以利用编程设备检查用户存储器内容、改变存储器内容、改变 PLC 的各种设置；开关拨到 TERM 时，不改变当前操作模式，此模式多数用于联网的 PLC 网络或现场调试。如果需要 CPU 模块上电时自动运行程序，模式开关必须在 RUN 位置。

2）将模式开关拨到 RUN 或 TERM 时，可以由 STEP7-Micro/WIN V4.0 编程软件控制 CPU 模块的运行和停止。

3）在程序中插入 STOP 指令，可以在条件满足时将 CPU 模块设置为停止模式。

5. 通信端口和扩展 I/O 端口

在 CPU 模块左侧的通信端口是连接编程器或其他外部设备的接口，S7-200 PLC 的通信端口为 RS485 端口。扩展 I/O 端口位于 CPU 模块右侧的前盖下面，如图 1.1-5 所示，它是连接各种扩展模块的接口。

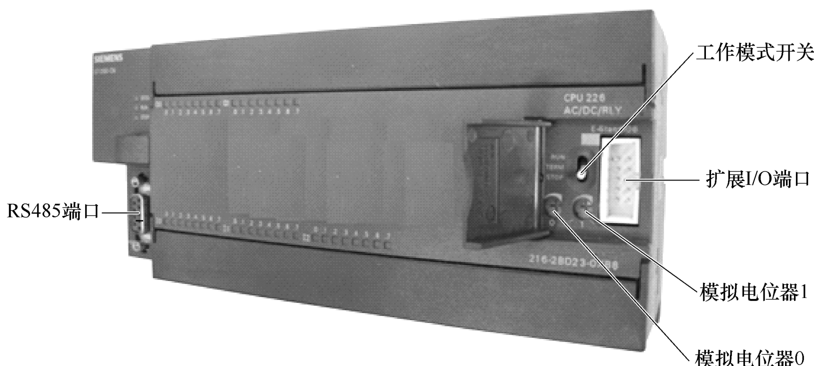


图 1.1-5　CPU226 模块的前盖下的布局

6. 模拟电位器

揭开 CPU 模块右侧的前盖就会看到一个或两个模拟电位器，如图 1.1-5 所示。调节这些电位器就会改变特殊存储器 SMB28 和 SMB29 这两个字节中的值，以改变程序运行时的参数，如定时器和计数器的预置值、过程量的控制参数。

7. 可选卡插槽与可选卡

在 CPU 模块的左侧有一个可选卡插槽。根据需要，可选卡插槽可以插入下述三种卡中

的一种：存储卡、电池卡、日期/时钟电池卡。

存储卡 MC291 提供 EEPROM 存储单元。在 CPU 模块上插入存储卡后，就可使用编程软件 STEP7-Micro/WIN V4.0 将 CPU 模块中的存储内容（系统块、程序块和数据块等）复制到卡上；或将存储卡插到其他 CPU 模块上，通电时存储卡中的内容会自动复制到 CPU 模块中。用存储卡传递程序时，被写入的 CPU 模块必须与提供程序来源的 CPU 模块相同或更高型号。

电池卡 BC293 为所有型号的 CPU 模块提供数据保持的后备电池，该电池在内置的超级电容放电完毕后起作用。

日期/时钟电池卡 CC292 用于 CPU221 和 CPU222 两种不具备内置时钟功能的 CPU 模块，以提供日期/时钟功能，同时提供后备电池。电池卡能够保持数据和内置时钟长达 200 天。

二、PLC 的基本结构

PLC 实质上是一种工业控制计算机，有着与通用计算机相类似的结构，PLC 也是由硬件和软件两大部分组成的。

1. PLC 硬件结构

PLC 硬件结构主要由中央处理器（CPU）、存储器、输入/输出单元（I/O 接口）、I/O 扩展接口、通信及编程接口、电源变换器等组成，见图 1.1-6 所示的点画线框内。

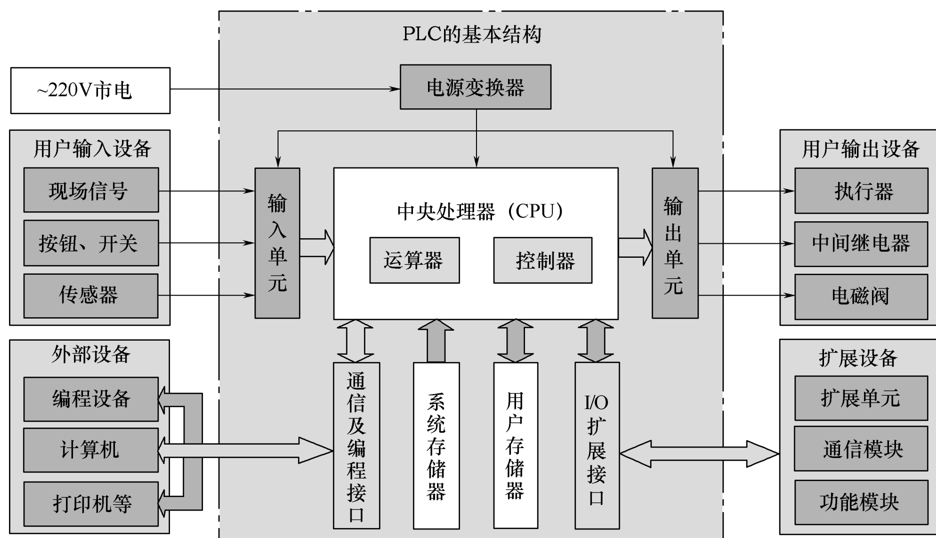


图 1.1-6　PLC 基本结构组成

（1）中央处理器（CPU）　CPU 是 PLC 的核心部件，由运算器和控制器组成。CPU 可以是通用微处理器、单片机或位片式微处理器。它通过控制总线、地址总线和数据总线与存储器、输入/输出单元和通信接口等建立联系。其主要用于接收并存储从编程器输入的用户程序，检查编程过程是否出错，进行系统诊断，解释并执行用户程序，完成通信及外设的某些功能。

（2）存储器　PLC 中的存储器主要有系统程序存储器、用户程序存储器以及工作数据存储器三种 。

1）系统程序存储器。其用于存放系统程序，这些程序在 PLC 出厂前就已经固化到只读存储器（ROM）中了。第一部分为系统管理程序，第二部分为用户指令解释程序，第三部

分为标准程序模块与系统调用程序。

2）用户程序存储器。其用于存储 PLC 用户的应用程序，在调试阶段，用户程序存放在读写存储器（RAM）中，可由备用电池（一般为锂电池）保存 2 ~3 年。

3）工作数据存储器 。工作数据存储器用来存储工作数据，即用户程序中使用的 ON/OFF 状态、数值数据等。

（3）输入/输出单元（I/O 接口） 输入/输出单元通常也称为输入/输出接口（I/O 接口)，是 PLC 与工业生产现场设备之间的连接部件。

1）输入接口。输入接口用来接收和采集用户输入设备产生的信号。输入信号主要有两种类型：一类是由按钮、选择开关、行程开关、继电器触点、接近开关、光电开关、数字拨码开关等来的开关量输入信号；另一类是由电位器、测速发电机和各种变送器等来的模拟量输入信号。这些信号经过光电隔离、滤波和电平转换等处理，变成 CPU 能够接收和处理的信号，并送给输入映像寄存器。

PLC 输入接口电路有直流输入、交流输入和交流/直流混合输入三种。输入接口的电源可以由外部提供，也可以由 PLC 内部提供。

图 1. 1-7 所示为西门子 S7-200 系列 PLC 的直流输入接口电路，图中只画出了对应于一个点的输入电路，各个输入点所对应的输入电路均相同。其中直流电源由外接提供，极性可以为任意极性。

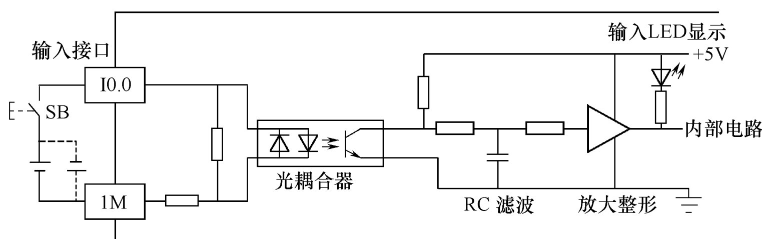


图 1. 1-7 S7-200 PLC 直流输入接口电路

2）输出接口。输出接口是将经过 CPU 处理的信号通过光电隔离和功率放大等处理，转换成外部设备所需要的驱动信号（数字量输出或模拟量输出)，以驱动外部各种执行设备，如接触器、指示灯、报警器、电磁阀、电磁铁、调节阀、调速装置等设备。

输出接口电路就是 PLC 的负载驱动回路。为适应实际设备控制的需要，输出接口的形式有继电器输出型、场效应晶体管输出型及双向晶闸管输出型三种，如图 1. 1-8 所示。为提高 PLC 抗干扰能力，每种输出电路都采用了光电或电气隔离技术。

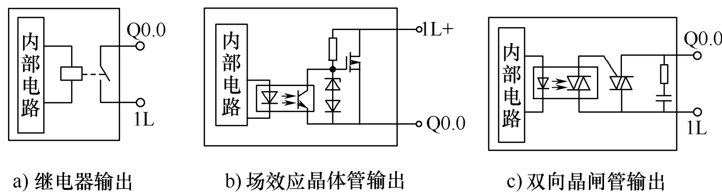


图 1. 1-8 输出接口的形式

图 1. 1-8a 所示继电器输出型为有触点的输出方式，既可以驱动直流负载，又可以驱动

交流负载，驱动负载的能力为 2A 左右。其优点是适用电压范围比较宽、导通压降小、承受瞬时过电压和过电流的能力强，缺点是动作速度较慢、响应时间长、动作频率低，建议在输出量变化不频繁时优先选用，不能用于高速脉冲的输出。其电路工作原理是：当内部电路的状态为“1”时，使继电器线圈通电，产生电磁吸力，触点闭合，则负载得电，同时点亮输出指示灯 LED（图 1. 1-8a 中负载、输出指示灯 LED 未画出），表示该路输出点有输出；当内部电路的状态为“0”时，使继电器的线圈无电流，触点断开，则负载断电，同时 LED 熄灭，表示该路输出点无输出。

图 1. 1-8b 所示的场效应晶体管输出形式，只可以驱动直流负载，驱动负载的能力是每一个输出点的最大允许控制电流为 750mA。其优点是可靠性强、执行速度快、寿命长，缺点是过载能力差，适用于高速（可达 20kHz）、小功率直流负载。其电路工作原理是：当内部电路的状态为“1”时，光耦合器导通，使晶体管饱和导通，场效应晶体管也饱和导通，则负载得电，同时点亮 LED（图 1. 1-8b 中负载、LED 未画出），表示该路输出点有输出；当内部电路的状态为“0”时，光耦合器断开，晶体管截止，场效应晶体管也截止，则负载失电，LED 熄灭，表示该路输出点无输出。图 1. 1-8b 中的稳压管用来抑制关断过电压和外部的浪涌电压，以保护场效应晶体管。

图 1-1-8c 所示的双向晶闸管输出形式，适合驱动交流负载，驱动负载的能力为 1A 左右。由于双向晶闸管和晶体管同属于半导体器件，所以优缺点与晶体管输出形式相似。双向晶闸管输出形式适用于高速、大功率交流负载。其电路工作原理是：当内部电路的状态为“1”时，发光二极管导通发光，双向二极管导通，给双向晶闸管施加了触发信号，无论外接电源极性如何，双向晶闸管均导通，负载得电，同时输出指示灯 LED 点亮（图 1. 1-8c 中负载、输出指示灯 LED 未画出），表示该路输出点有输出；当内部电路的状态为“0”时，双向晶闸管无触发信号，双向晶闸管关断，此时负载失电，LED 熄灭，表示该路输出点无输出。

（4）扩展接口　扩展接口用来扩展 PLC 的 I/O 端子数。当用户所需要的 I/O 端子数超过 PLC 基本单元（即主机，带 CPU）的 I/O 端子数时，可通过此接口用扁平电缆线将 I/O 扩展接口（不带有 CPU）与 PLC 基本单元相连接，以增加 PLC 的 I/O 端子数，从而适应控制系统的要求。其他很多的智能单元也通过该接口与 PLC 基本单元相连。

（5）通信接口　通信接口是专用于数据通信的，主要实现人-机对话。PLC 通过通信接口可与打印机、监视器以及其他的 PLC 或计算机等设备实现通信。

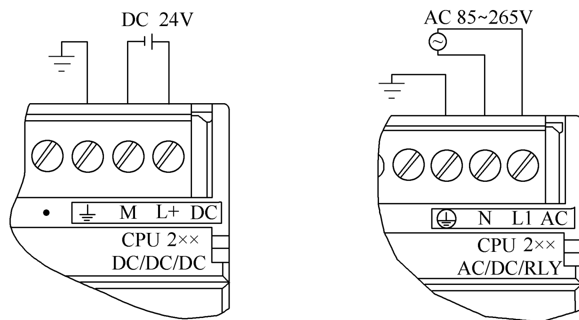


图 1. 1-9　PLC 供电形式

（6）电源　PLC 的电源（见图 1. 1-9）是指将外部输入的电源处理后转换成满足 PLC 的 CPU、存储器、输入/输出接口等内部电路工作需要的直流 5V 电源电路或电源模块。另一方面可为外部输入元件提供 DC 24V 标准电源，而驱动 PLC 负载的电源由用户提供。

2. PLC 软件

PLC 软件由系统程序和用户程序组成。

（1）系统程序　系统程序是由 PLC 制造厂商采用汇编语言设计编写的，固化于 ROM 型系统程序存储器中，用于控制 PLC 本身的运行，用户不能直接读写与更改。系统程序分为系统管理程序、用户指令解释程序、标准程序模块和系统调用程序。

（2）用户程序　用户程序是用户为完成某一控制任务而利用 PLC 的编程语言编制的程序。由于 PLC 是专门为工业控制而开发的装置，其主要使用者是广大电气技术人员，为了满足他们的传统习惯和掌握能力，PLC 的编程语言采用比计算机语言相对简单、易懂、形象的专用语言。PLC 的主要编程语言有梯形图和语句表等。

三、PLC 的工作原理

1. PLC 的工作过程

PLC 在本质上虽然是一台微型计算机，其工作原理与普通计算机类似，但是 PLC 的工作方式却与计算机有很大的不同。计算机一般采用等待输入—响应（运算和处理）—输出的工作方式，如果没有输入，就一直处于等待状态；而 PLC 采用的是周期性循环扫描的工作方式，每一个周期要按部就班做完全相同的工作，与是否有输入或输入是否变化无关。

PLC 的工作过程一般包括内部处理、通信操作、输入处理、程序执行、输出处理五个阶段，如图 1. 1-10 所示。

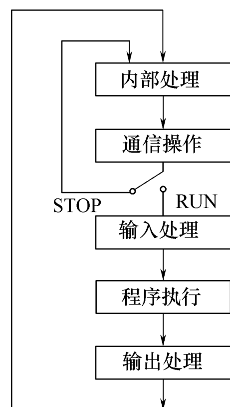


图 1. 1-10　PLC 的工作过程

（1）内部处理　PLC 检查 CPU 模块内部的硬件是否正常，进行监控、定时器复位等工作。在运行模式下，还要检查用户程序存储器，如果发现异常，则停止并显示错误；若自诊断正常，继续向下扫描。

（2）通信操作　在通信操作阶段，CPU 自检并处理各通信端口接收到的任何信息，完成数据通信服务，即检查是否有计算机、编程器的通信请求，若有则进行相应处理。

（3）输入处理　输入处理阶段又称输入采样阶段。在此阶段，按顺序扫描输入端子，把所有外部输入电路的接通/断开状态读入到输入映像寄存器，输入映像寄存器被刷新。

（4）程序执行　用户程序在 PLC 中是顺序存放的。在程序执行阶段，在无中断或跳转指令的情况下，CPU 根据用户程序从第一条指令开始按自上而下、从左至右的顺序逐条扫描执行。

（5）输出处理　当所有指令执行完毕后，进入输出处理阶段，又称输出刷新阶段。CPU 将输出映像寄存器中的内容集中转存到输出锁存器，然后传送到各相应的输出端子，最后再驱动外部负载。

【工作模式解读】

PLC 有两种工作模式，即运行（RUN）模式和停止（STOP）模式。运行模式是执行应用程序的过程，停止模式一般用于程序的编制与修改。

当 PLC 工作方式开关置于 RUN 时，执行所有阶段；当 PLC 工作方式开关置于 STOP 时，不执行后三个阶段，此时可进行通信操作、对 PLC 编程等。

2. PLC 用户程序的执行过程

在运行模式下，PLC 对用户程序重复地执行输入处理、程序执行、输出处理三个阶段，如图 1.1-11 所示，图中的序号表示梯形图程序的执行顺序。

在用户程序执行过程中，输入映像寄存器的内容，由上一个输入采样期间输入端子的状态决定。输出映像寄存器的状态，由程序执行期间的执行结果所决定，随程序执行过程而变化。输出锁存器的状态，由程序执行期间输出映像寄存器的最后状态来确定。各输出端子的状态，由输出锁存器确定。程序如何执行，取决于输入、输出映像寄存器的状态。

在每次扫描中，PLC 只对输入采样一次，输出刷新一次，这可以确保在程序执行阶段，同一个扫描周期的输入映像寄存器和输出锁存器中的内容保持不变。每重复一次的时间就是一个扫描周期，其典型值为 1～100ms。扫描周期与用户程序的长短、指令的种类和 CPU 执行指令的速度有很大的关系。

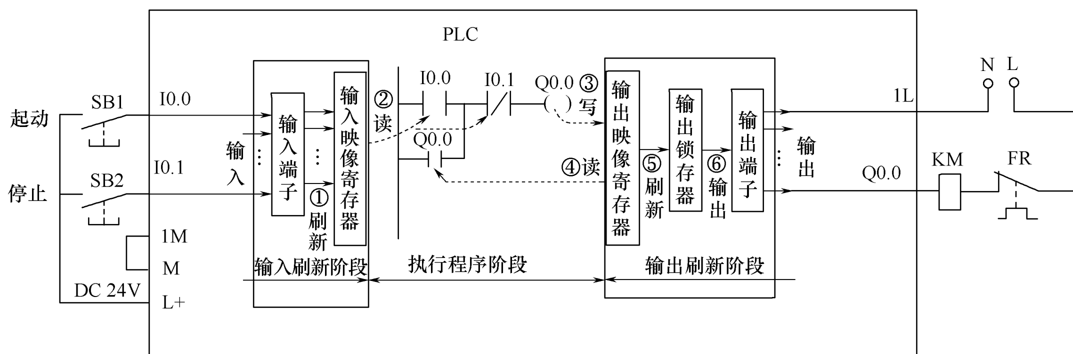


图 1.1-11　PLC 用户程序的执行过程

四、PLC 的特点与分类

西门子 S7 系列 PLC 分为 S7-400、S7-300、S7-200 三个系列，分别为 S7 系列的大、中、小型 PLC。S7-200 系列 PLC 包括 CPU21X 型以及 CPU22X 型。其中 CPU22X 型 PLC 提供了四个不同的基本型号，即 CPU221、CPU222、CPU224 和 CPU226。下面重点介绍 S7-200 小型机。

小型 PLC 中，CPU221 价格低廉且能满足多种集成功能的需要。CPU222 是 S7-200 家族中低成本的单元，通过可连接的扩展模块即可处理模拟量。CPU224 具有更多的输入/输出点及更大的存储器。CPU226 和 CPU226XM 是该系列中功能最强的单元，可完全满足一些中小型复杂控制系统的要求。四种型号的 PLC 具有下列特点。

（1）集成的 24V 电源　其可直接连接到传感器、变送器和执行器。CPU221 和 CPU222 具有 180mA 输出，CPU224 输出 280mA，CPU226、CPU226XM 输出 400mA 可用作负载电源。

（2）高速脉冲输出　PLC 具有两路高速脉冲输出端，输出脉冲频率可达 20kHz，用于控制步进电动机或伺服电动机，实现定位任务。

（3）通信口　CPU221、CPU222 和 CPU224 具有一个 RS485 通信口，CPU226、CPU226XM 具有两个 RS485 通信口。支持 PPI、MPI 通信协议，有自由口通信能力。

（4）模拟电位器　CPU221/222 有一个模拟电位器，CPU224/226/226XM 有两个模拟电

位器。模拟电位器用来改变特殊寄存器（SMB28、SMB29）中的数值，以改变程序运行时的参数，如定时器、计数器的预置值以及过程量的控制参数。

（5）中断输入允许以极快的速度对过程信号的上升沿做出响应。

（6）EEPROM 存储器模块（选件） 可作为修改与复制程序的快速工具，无需编程器并可进行辅助软件归档工作。

（7）电池模块 用户数据（如标志位状态、数据块、定时器、计数器）可通过 PLC 内部的超级电容存储大约 5 天，选用电池模块能延长存储时间到 200 天（10 年寿命）。电池模块插在存储器模块的卡槽中。

（8）不同的设备类型 CPU221～226 各有两种类型 CPU，具有不同的电源电压和控制电压。

（9）数字量输入/输出点 CPU221 具有 6 个输入点和 4 个输出点，CPU222 具有 8 个输入点和 6 个输出点，CPU224 具有 14 个输入点和 10 个输出点，CPU226/226XM 具有 24 个输入点和 16 个输出点。CPU22X 主机的输入点为 24V 直流双向光耦合输入电路，输出有继电器和直流（MOS 型）两种类型。

（10）高速计数器 CPU221/222 有 4 个 1～50kHz 高速计数器，CPU224/226/226XM 有 6 个 1～50kHz 高速计数器，用于捕捉比 CPU 扫描频率更快的脉冲信号。

第二节 S7-200 PLC 的编程元件及语言

一、基本数据类型与寻址方式

1. S7-200 系列 PLC 的存储器区域

S7-200 的存储器分为用户程序空间、CPU 组态空间和数据区空间。

用户程序空间用于存放用户程序，存储器为 EEPROM；CPU 组态空间用于存放有关 PLC 配置结构参数，如 PLC 主机及扩展模块的 I/O 配置和编址、配置 PLC 站地址、设置保护口令、停电记忆保持区、软件滤波功能等，存储器为 EEPROM；数据区空间是用户程序执行过程中的内部工作区域，该区域存放输入信号、运算输出结果、计时值、计数值、高速计数值和模拟量数值等，存储器为 EEPROM 和 ROM。

数据区空间是 S7-200 CPU 提供的存储器的特定区域，数据区空间使 CPU 的运行更快、更可靠。S7-200 系列 PLC 的数据存储区按存储器存储数据的长短可划分为字节存储器、字存储器和双字存储器三类。字节存储器有七个，即输入映像寄存器（I）、输出映像寄存器（Q）、变量存储器（V）、位存储器（M）、特殊存储器（SM）、顺序控制继电器（S）、局部变量存储器（L）；字存储器有四个，即定时器（T）、计数器（C）、模拟量输入映像寄存器（AI）和模拟量输出映像寄存器（AQ）；双字存储器有两个，即累加器（AC）和高速计数器（HC）。

用户对用户程序空间、CPU 组态空间和部分数据区空间进行编辑，编辑后写入 PLC 的 EEPROM。RAM 为 EEPROM 存储器提供备份存储区，用于 PLC 运行时动态使用。RAM 由大容量电容做停电保持。

2. 数据区空间存储器的编址方式

在计算机中使用的数据均为二进制数，二进制数的基本单位是 1 个二进制位，8 个二进制位组成 1 个字节，2 个字节组成 1 个字，2 个字组成 1 个双字。

存储器是由许多存储单元组成的，每个存储单元都有唯一的地址，可以依据存储器地址来存取数据。数据区空间存储器的单位可以是位、字节、字、双字，编址方式也可以是位编址、字节编址、字编址和双字编址。

（1）位编址 位编址：存储器标识符 + 字节地址 + 位地址，如 I0.1、M0.0、Q0.3 等。如图 1.2-1 所示，I1.4 表示图中黑色标记的位地址，I 是输入映像寄存器的区域标识符，1 是字节地址，4 是位地址，在字节地址和位地址之间用点号“.”隔开。

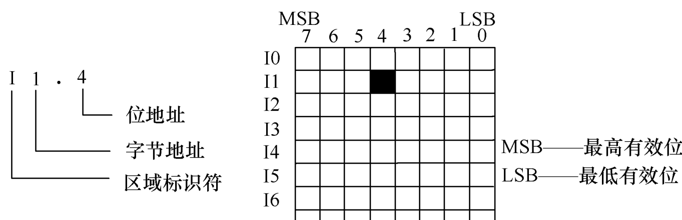


图 1.2-1 位编址 I1.4 的表达方式

按照这种位编址方式编址的存储区有输入映像寄存器（I）、输出映像寄存器（Q）、位存储器（M）、特殊存储器（SM）、局部变量存储器（L）、变量存储器（V）和顺序控制继电器（S）。

（2）字节编址、字编址和双字编址 字节、字、双字的编址方式如图 1.2-2 所示。

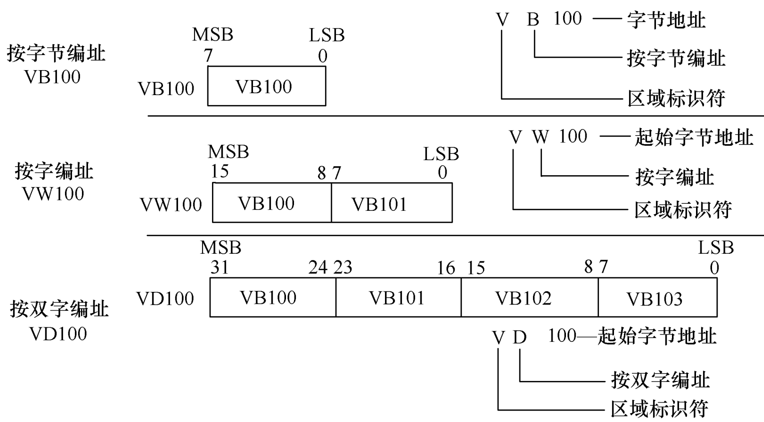


图 1.2-2 字节、字、双字的编址方式

字节编址：存储器标识符 + 字节长度（B）+ 字节地址，如 IB0、QB0、VB100 等。

字编址：存储器标识符 + 字长度（W）+ 起始字节地址，如 VW100 表示 VB100、VB101 这两个字节组成的字，其中 VB100 是高有效字节，VB101 是低有效字节。

双字编址：存储器标识符 + 双字长度（D）+ 起始字节地址，如 VD100 表示由 VW100、VW102 这两个字组成的双字或由 VB100、VB101、VB102、VB103 这四个字节组成的双字，其中 VB100 是最高有效字节，VB103 是最低有效字节。

按照这种字节、字和双字编址方式编址的存储区有输入映像寄存器（I）、输出映像寄存器（Q）、位存储器（M）、特殊存储器（SM）、局部变量存储器（L）、变量存储器（V）、顺序控制继电器（S）、模拟量输入映像寄存器（AI）和模拟量输出映像寄存器（AQ）。

（3）其他编址方式　数据区空间存储器区域中还包括定时器、计数器、累加器、高速计数器等，它们是模拟相关的电器元件，编址方式为区域标识符 + 元件号。例如，T24 表示某定时器的地址，T 是定时器的区域标识符，24 是定时器号。

3. S7-200 的基本数据类型

在 S7-200 的编程语言中，大多数指令要与数据对象一起进行操作。不同的数据对象具有不同的数据类型，不同的数据类型又具有不同的数制和格式选择。因此，程序中所使用的数据需要指定一种数据类型，而在指定数据类型时，首先要确定数据大小及数据位的结构。S7-200 的基本数据类型及其范围见表 1.2-1。

表 1.2-1　S7-200 的基本数据类型及其范围

基本数据类型		位　数	说　明
布尔型 BOOL		1	位范围：0，1
无符号数	字节型 BYTE	8	字节范围：0 ~ 255
	字型 WORD	16	字范围：0 ~ 65535
	双字型 DWORD	32	双字范围：0 ~（$2^{32}-1$）
有符号数	字节型 BYTE	8	字节范围：-128 ~ +127
	整型 INT	16	整数范围：-32768 ~ +32767
	双整型 DINT	32	双字整数范围：-2^{31} ~（$2^{31}-1$）
实数型 REAL		32	IEEE 浮点数

编程中经常会使用常数，常数数据长度可分为字节、字和双字。在机器内部的数据都以二进制形式存储，但常数的书写可以用二进制、十进制、十六进制、ASCII 码或浮点数（实数）等多种形式。几种常数形式说明如下：

1）二进制的书写格式为“2#二进制数值”，如 2#0101_ 1100_ 0010_ 1010；

2）十进制的书写格式为“十进制数值”，如 1052；

3）十六进制的书写格式为“16#十六进制数值”，如 16#8AC6；

4）ASCII 码的书写格式为“‘ASCII 码文本’”，如‘good bye’；

5）浮点数的书写格式按 IEEE 浮点数格式，如 I0.5。

4. S7-200 CPU 模块操作数的数值范围

S7-200 CPU 模块操作数的数值范围见表 1.2-2。

表 1.2-2　S7-200 CPU 模块操作数的数值范围

存取方式	CPU 221	CPU 222	CPU 224	CPU 224XP CPU 224XP si	CPU 226
位存取 （字节、位）	I0.0 ~ I15.7　Q0.0 ~ Q15.7　M0.0 ~ M31.7　T0 ~ T255　C0 ~ C255　L0.0 ~ L63.7　S0.0 ~ S31.7				
	V0.0 ~ V2047.7		V0.0 ~ 8191.7	V0.0 ~ V10 239.7	
	SM0.0 ~ SM165.7	SM0.0 ~ SM299.7	SM0.0 ~ SM549.7		

（续）

<table>
<tr><th>存 取 方 式</th><th>CPU 221</th><th>CPU 222</th><th>CPU 224</th><th>CPU 224XP
CPU 224XP si</th><th>CPU 226</th></tr>
<tr><td rowspan="4">字节存取</td><td colspan="5">IB0 ~ IB15　QB0 ~ QB15　MB0 ~ MB31　SB0 ~ SB31　LB0 ~ LB63</td></tr>
<tr><td colspan="3">AC0 ~ AC3</td><td colspan="2">AC0 ~ AC255</td></tr>
<tr><td colspan="2">VB0 ~ VB2 047</td><td>VB0 ~ VB8 191</td><td colspan="2">VB0 ~ VB10 239</td></tr>
<tr><td>SMB0 ~ SMB165</td><td>SMB0 ~ SMB299</td><td colspan="3">SMB0 ~ SMB549</td></tr>
<tr><td rowspan="4">字存取</td><td colspan="5">IW0 ~ IW14　QW0 ~ QW14　MW0 ~ MW30　SW0 ~ SW30　T0 ~ T255　C0 ~ C255　LW0 ~ LW58　AC0 ~ AC3</td></tr>
<tr><td colspan="2">VW0 ~ VW2 046</td><td>VW0 ~ VW8 190</td><td colspan="2">VW0 ~ VW10 238</td></tr>
<tr><td>SMW0 ~ SMW164</td><td>SMW0 ~ SMW298</td><td colspan="3">SMW0 ~ SMW548</td></tr>
<tr><td colspan="2">AIW0 ~ AIW30　AQW0 ~ AQW30</td><td colspan="3">AIW0 ~ AIW62　AQW0 ~ AQW62</td></tr>
<tr><td rowspan="3">双字存取</td><td colspan="5">ID0 ~ ID12　QD0 ~ QD12　MD0 ~ MD28　SD0 ~ SD28　LD0 ~ LD60　AC0 ~ AC3　HC0 ~ HC5</td></tr>
<tr><td colspan="2">VD0 ~ VD2 044</td><td>VD0 ~ VD8 188</td><td colspan="2">VD0 ~ VD10 236</td></tr>
<tr><td>SMD0 ~ SMD162</td><td>SMD0 ~ SMD296</td><td colspan="3">SMD0 ~ SMD546</td></tr>
</table>

二、PLC 的编程元件

PLC 在其系统软件的管理下，将用户程序存储器（即装载存储区）划分出若干个区，并赋予这些区不同的功能，分别称为输入继电器、输出继电器、辅助继电器、变量寄存器、定时器、计数器、数据寄存器等。

说明：在 PLC 内部，并不真正存在这些实际的物理器件，与其对应的只是存储器中的某些存储单元。

1. 输入继电器 I

输入继电器 I 就是 PLC 存储系统中的输入映像寄存器。它通过输入继电器，将 PLC 的存储系统与外部输入端子建立明确的关系，一般按“字节 . 位”的编址方式来读取一个继电器的状态。S7-200 PLC（CPU224XP）提供了 128 个输入映像寄存器，编址范围为 I0. 0 ~ I15. 7。

2. 输出继电器 Q

输出继电器 Q 就是 PLC 存储系统中的输出映像寄存器。它通过输出继电器，将 PLC 的存储系统与外部输出端子建立明确的关系，一般按“字节 . 位”的编址方式来读取一个继电器的状态。S7-200 PLC（CPU224XP）提供了 128 个输出映像寄存器，编址范围为 Q0. 0 ~ Q15. 7。

3. 变量寄存器 V

S7-200 PLC 中有大量的变量寄存器，用于模拟量控制、数据运算、参数设置及存放程序执行过程中控制逻辑操作的中间结果。其数量与 CPU 型号有关，CPU222 为 V0. 0 ~ V2047. 7、CPU224 为 V0. 0 ~ V5119. 7、CPU226 为 V0. 0 ~ V5119. 7。

4. 辅助继电器 M

辅助继电器 M 的功能与传统的继电器控制电路中的中间继电器相同，它借助于辅助继

电器的编程，可使输入/输出之间建立复杂的逻辑关系和联锁关系，以满足不同的控制要求。在 S7-200 PLC（CPU224XP）中辅助继电器 M 的个数为 256 个，编址范围为 M0.0 ~ M31.7。

5. 特殊继电器 SM

特殊继电器 SM 用来存储系统的状态变量及有关的控制参数和信息。用户可以通过特殊继电器向 PLC 反映对操作的特殊要求以及沟通 PLC 与被控对象之间的信息，PLC 通过特殊继电器向用户提供一些特殊的控制功能和系统信息。S7-200 PLC（CPU224XP）提供了2 400 个特殊继电器即 SM0.0 ~ SM299.7，分为只读型和读写型两类，其中只读型的 30 个特殊继电器为 SM0.0 ~ SM29.7。

例如：

SM0.0：运行监控，PLC 在运行状态时，SM0.0 总为 ON。

SM0.1：初始脉冲，PLC 由 STOP 转为 RUN 时，ON 一个扫描周期。

SM0.3：PLC 上电进入运行状态时，ON 一个扫描周期。

SM0.4：分时钟脉冲，占空比为 50%，周期为 1min 的脉冲串。

SM0.5：秒时钟脉冲，占空比为 50%，周期为 1s 的脉冲串。

SM0.6：扫描时钟，一个周期为 ON，下个周期为 OFF，交替循环。

SMB28 和 SMB29 ：分别对应模拟电位器 0 和 1 的当前值，数值范围为 0 ~ 255。

6. 定时器 T

定时器 T 是 PLC 的重要编程元件，它的作用与继电器控制电路中的时间继电器基本相似，用来实现按照时间原则进行控制的目的。定时器的设定值通过程序预先输入，当满足定时器的工作条件时，定时器开始定时，当前值从 0 开始增加；当达到设定值时定时器动作，其动合触点和动断触点动作。

S7-200 PLC（CPU224XP）中定时器的数量为 256 个，范围为 T0 ~ T255。它分为三种类型：接通延时定时器 TON、断开延时定时器 TOF 及保持型接通延时定时器 TONR。

7. 计数器 C

计数器 C 的作用是对编程元件状态脉冲的上升沿进行积累计数，从而实现计数操作。当条件满足时，计数器开始计数，当前值达到设定值后，计数器的动合触点和动断触点动作，实现计数操作。

S7-200 PLC（CPU224XP）中计数器的数量为 256 个，范围为 C0 ~ C255。它分为三种类型：递增计数、递减计数和增/减计数。

8. 状态（顺序控制）继电器 S

状态继电器 S 又称状态元件，是使用步进控制指令编程时的重要编程元件，用来组织机器操作或进入等效程序段工步，以实现顺序控制和步进控制。顺序控制继电器采用顺序功能图法编程，每一个状态继电器可以用来代表控制状态中的一个步序（能为编程提供方便），可以按位、字节、字或双字来存取 S 位。S7-200 PLC（CPU224XP）提供了 256 个状态继电器，编址范围为 S0.0 ~ S31.7。

三、PLC 的编程语言

PLC 为用户提供了完整的编程语言，以适应编制用户程序的需要。PLC 提供的编程语言通常有梯形图（LAD）、指令表（ST）、顺序功能流程图（SFC）和功能块图（FBD）等几

种，下面以 S7-200 系列 PLC 为例加以介绍。

1. 梯形图

梯形图（LAD）是国内使用得最多的图形编程语言，称为 PLC 的第一编程语言。它沿用了电气工程师熟悉的传统继电器控制电路图的形式和概念，其基本控制思想与继电器控制电路图很相似，只是在使用符号和表达方式上有一定区别。图 1.2-3 为一个典型的梯形图。

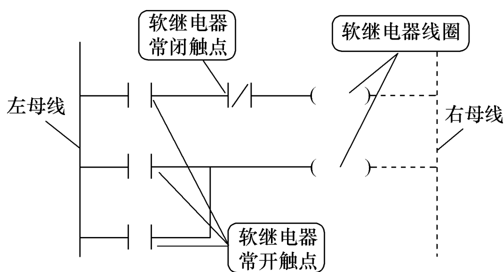


图 1.2-3　PLC 梯形图

梯形图是由两条母线（左右两条垂直的线）和两母线之间的逻辑触点和线圈按一定结构形式连接起来类似于梯子的图形（梯形图常称为电路或程序，梯形图的设计称为编程）。梯形图直观易懂，很容易掌握，为了更好地理解梯形图，这里把 PLC 与继电器控制电路相对比做介绍，重点理解几个与梯形图相关的概念。

表 1.2-3 给出了 PLC 与继电器控制电路的电气符号对照关系。

表 1.2-3　PLC 与继电器控制电路中的电气符号对照

触点、线圈	继电器符号	PLC 符号
常开触点		
常闭触点		
线圈		

（1）软继电器（即映像寄存器）　PLC 梯形图中的某些编程元件沿用了继电器这一名称，如输入继电器、输出继电器、内部辅助继电器等，但是它们不是真实的物理继电器，而是一些存储单元（软继电器），每一软继电器与 PLC 存储器中映像寄存器的一个存储单元相对应。该存储单元如果为“1”状态，则表示梯形图中对应软继电器的线圈“通电”，其常开触点接通，常闭触点断开，称这种状态为该软继电器的“1”或“ON”状态。如果该存储单元为“0”状态，对应软继电器的线圈和触点的状态与上述相反，称该软继电器为“0”或“OFF”状态。使用中也常将这些“软继电器”称为编程元件。

（2）能流　当触点接通时，有一个假想的“概念电流”或“能流”从左向右流动，这一方向与执行用户程序时的逻辑运算顺序是一致的。能流只能从左向右流动。利用能流这一概念，可以帮助人们更好地理解和分析梯形图。

（3）母线　梯形图两侧的垂直公共线称为母线。在分析梯形图的逻辑关系时，为了借用继电器电路图的分析方法，可以想象左右两侧母线（左母线和右母线）之间有一个左正右负的直流电源电压，母线之间有“能流”从左向右流动。右母线可以不画出。

（4）梯形图的逻辑运算　根据梯形图中各触点的状态和逻辑关系，求出与图中各线圈对应的编程元件的状态，称为梯形图的逻辑运算。梯形图中逻辑运算是按从左至右、从上到下的顺序进行的。运算的结果马上可以被后面的逻辑运算所利用。逻辑运算是根据输入映像

寄存器中的值，而不是根据运算瞬时外部输入触点的状态来进行的。

2. 指令表

指令表（STL）编程语言类似于计算机中的助记符语言，它是 PLC 最基础的编程语言。所谓指令表编程，是用一个或几个容易记忆的字符来代表 PLC 的某种操作功能。图 1.2-4 所示为一个简单的 PLC 程序，图 a 是梯形图程序，图 b 是相应的指令表。一般来说，指令表编程适合于熟悉 PLC 和有经验的程序员使用。

3. 顺序功能流程图

顺序功能流程图（SFC）编程是一种图形化的编程方法，亦称功能图，如图 1.2-5 所示。使用它可以对具有并行、选择等复杂结构的系统进行编程，许多 PLC 都提供了用于 SFC 编程的指令。目前，国际电工委员会（IEC）也正在实施并发展这种语言的编程标准。

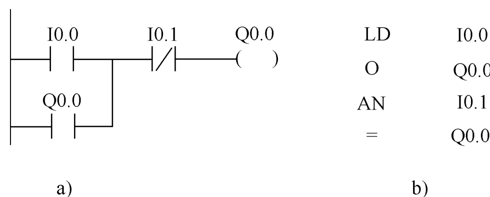


图 1.2-4　一个简单的 PLC 程序

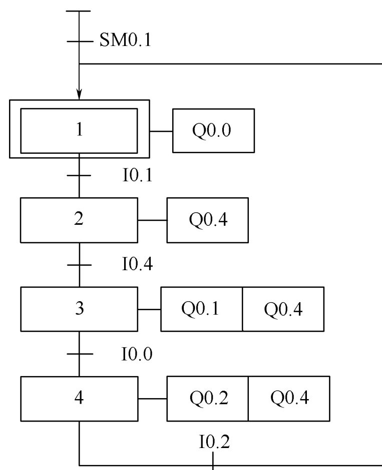


图 1.2-5　顺序功能流程图

4. 功能块图

S7-200 系列 PLC 专门提供了功能块图（FBD）编程语言，利用 FBD 可以查看到像普通逻辑门图形的逻辑盒指令。它没有梯形图编程器中的触点和线圈，但有与之等价的指令，这些指令是作为盒指令出现的，程序逻辑由这些盒指令之间的连接决定。也就是说，一条指令（如 AND 盒）的输出可以用来允许另一条指令（如定时器）的执行，这样可以建立所需要的控制逻辑。这样的连接思想可以解决范围广泛的逻辑问题。FBD 编程语言有利于程序流的跟踪，但目前使用较少。图 1.2-6 所示为 FBD 的一个简单实例。

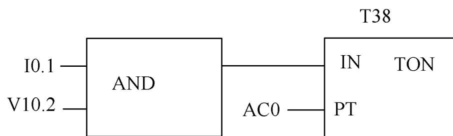


图 1.2-6　功能块图

第三节　STEP7-Micro/WIN V4.0 编程软件的安装与操作

由于 PLC 类型较多，不同机型对应的编程软件存在一定的差别，特别是不同厂家的 PLC 之间，它们的编程软件不能通用。STEP7-Micro/WIN V4.0 编程软件是基于 Windows 的

应用软件，由西门子公司专为 SIMATIC S7-200 系列 PLC 研制开发。该软件功能强大，界面友好，有联机帮助功能，既可用于开发用户程序，又可实时监控用户程序的执行状态。

一、系统安装要求

运行 STEP7-Micro/WIN V4.0 编程软件的计算机系统要求如下。

计算机配置：IBM486 以上兼容机，内存 8MB 以上，VGA 显示器，至少 50MB 以上硬盘空间。

操作系统：Windows 95 以上的操作系统。

二、软件安装步骤

STEP7-Micro/WIN V4.0 编程软件可以从西门子公司网站（www.ad.siemens.com.cn）免费下载，也可以用光盘安装，安装步骤如下。

1）双击 STEP7-Micro/WIN V4.0 的安装程序 setup.exe，开始安装。

2）出现“选择设置语言”对话框，选择“英语”选项，单击“确定”按钮，如图 1.3-1所示。

3）出现安装向导界面，如图 1.3-2 所示，单击“Next”按钮。

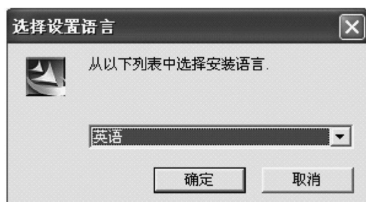


图 1.3-1　选择设置语言



图 1.3-2　确认安装

4）出现安装向导界面，如图 1.3-3 所示，单击“Yes”按钮。

5）出现安装向导界面，如图 1.3-4 所示，选择安装路径及名称，确认后单击“Next”按钮。该文件夹为今后的默认路径。

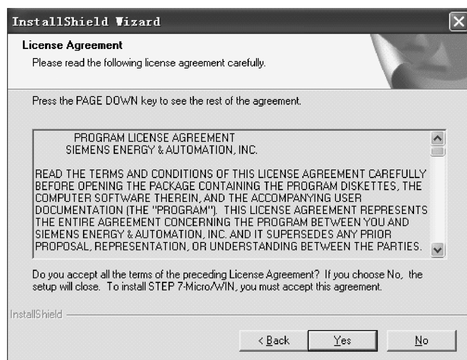


图 1.3-3　接受安装协议

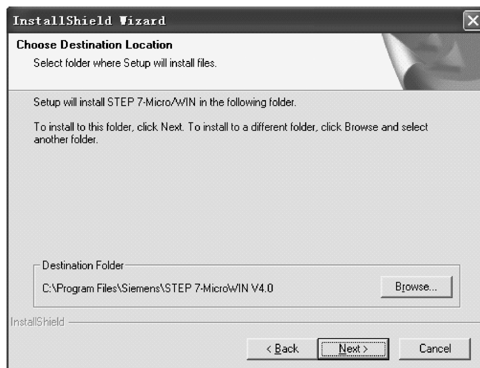


图 1.3-4　选择安装目录

6）安装进行中，如图 1.3-5 所示。

7）出现“Set PG/PC Interface”对话框，如图 1.3-6 所示。确认“PC/PPI cable (PPI)”后，单击“OK”按钮。

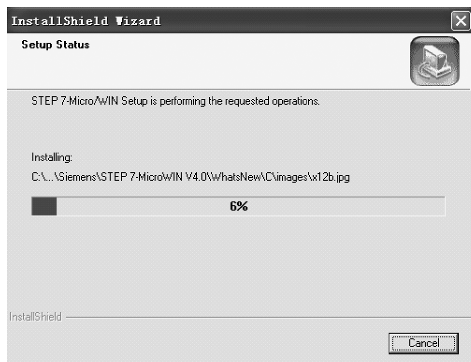


图 1.3-5　安装进行中

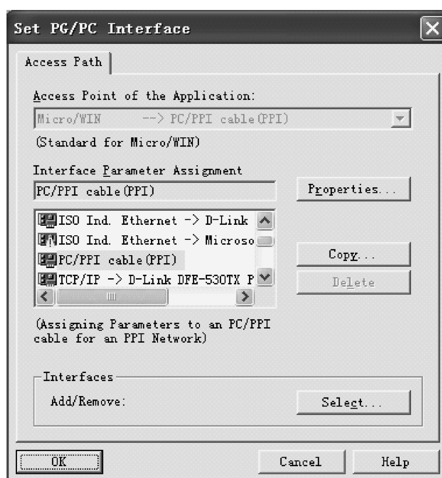


图 1.3-6　选择通信功能

8）安装完成，如图 1.3-7 所示。

9）默认选项是“Yes，I want to restart my computer now”，建议用户选择默认选项，单击“Finish”按钮，自动重新启动计算机，以完成程序的安装。重新启动 Windows 后，桌面上增加了相关软件的快捷方式，表明已经成功安装了 STEP7-Micro/WIN V4.0 软件。双击桌面上 STEP7-Micro/WIN V4.0 软件的快捷方式图标，则出现如图 1.3-8 所示的 STEP7-Micro/WIN V4.0 初始英文界面。

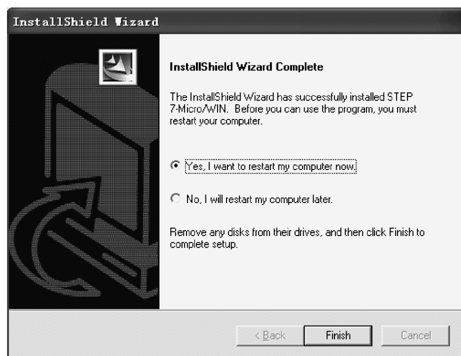


图 1.3-7　安装完成界面

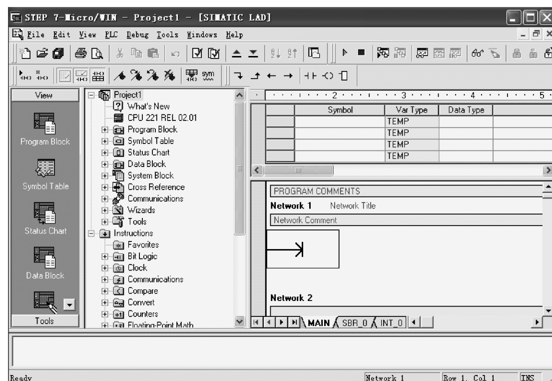


图 1.3-8　STEP7-Micro/WIN V4.0 初始英文界面

10）转换成中文界面，可以选择“Tools”→“Options”，在 Options 树中选择“General”，在 General 选项卡的 Language 列表中选择“Chinese”，如图 1.3-9 所示，单击“OK”按钮，根据提示做完相关步骤，则会退出初始英文界面。

11）重新打开桌面上的 STEP7-Micro/WIN V4.0 编程软件，就会出现 STEP7-Micro/WIN

V4.0 编程软件的中文界面，如图 1.3-10 所示。

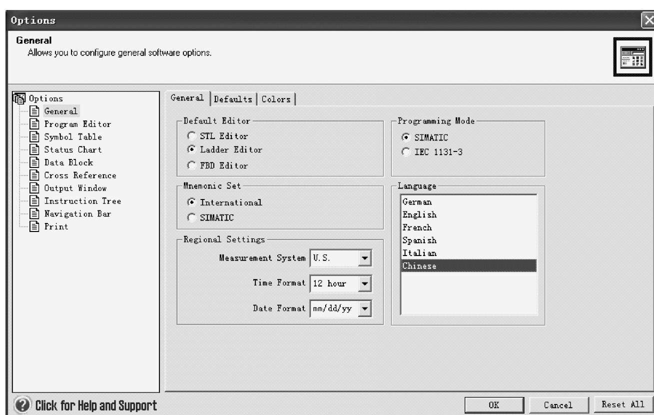


图 1.3-9　设置所需要的语言

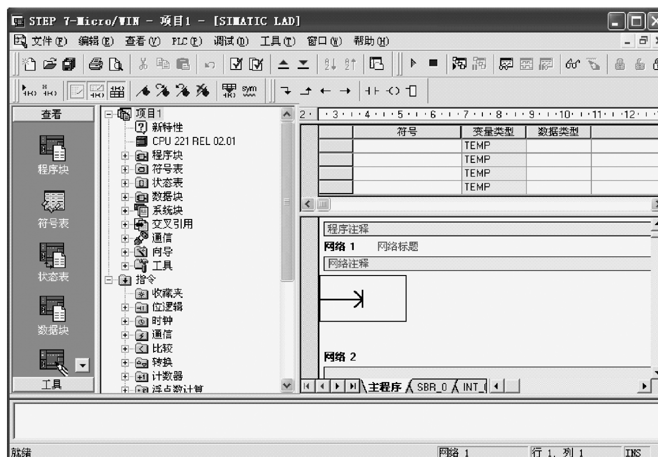


图 1.3-10　STEP7-Micro/WIN V4.0 编程软件的中文界面

三、认识 STEP7-Micro/WIN V4.0 编程软件主界面

点击桌面上 STEP 7--Micro/WIN 4.0 的图标 

，打开一个新的项目，显示如图 1.3-11所示的 STEP7-Micro/WIN V4.0 编程软件主界面。

STEP7-Micro/WIN V4.0 编程软件主界面一般可以分成以下几个区：标题栏、菜单栏（包含 8 个主菜单项）、工具栏（快捷按钮）、浏览条（快捷操作窗口）、指令树（快捷操作窗口）、输出窗口、状态栏和用户窗口（可同时或分别打开 5 个用户窗口）。除菜单栏外，用户可以根据需要决定其他窗口的取舍和样式。

1. 菜单栏

菜单栏共有 8 个主菜单选项，允许使用鼠标单击或采用按键执行操作各种命令，还可以定制“工具”菜单，在该菜单中增加命令和工具。菜单栏中各单项功能如下：

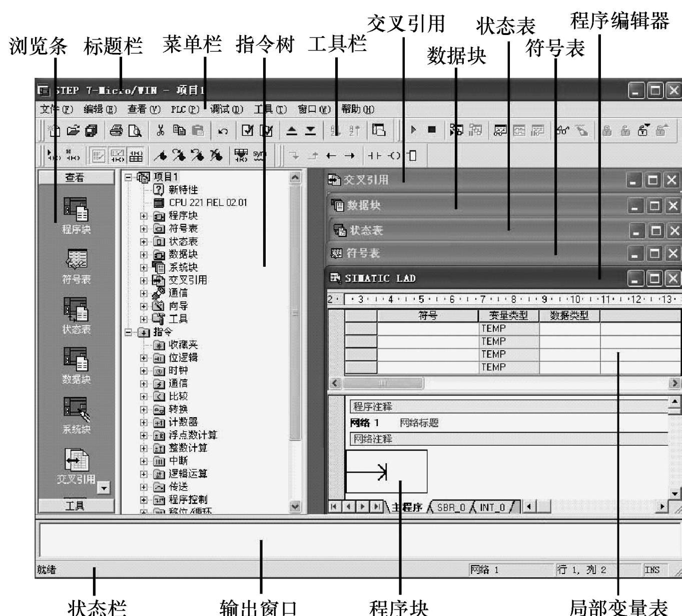


图 1.3-11　STEP7-Micro/WIN V4.0 编程软件主界面

1）文件（File）　菜单项可完成如新建、打开、关闭、保存文件，导入和导出、上传和下载程序，文件的页面设置、打印预览和打印设置等操作。

2）编辑（Edit）　菜单项提供编辑程序用的各种工具，如选择、剪切、复制、粘贴程序块或数据块的操作，以及查找、替换、插入、删除和快速光标定位等功能。

3）查看（View）　菜单项可以设置编程软件的开发环境，如打开和关闭其他辅助窗口（如引导窗口、指令树窗口、工具栏按钮区），执行引导条窗口的所有操作项目，选择不同语言的编程器（LAD、STL 或 FBD），设置三种程序编辑器的风格（如字体、指令盒的大小等）。

4）可编程序控制器（PLC）　菜单项用于实现与 PLC 联机时的操作，如改变 PLC 的工作方式、在线编译、清除程序和数据、查看 PLC 的信息，以及 PLC 的类型选择和通信设置等。

5）调试（Debug）　菜单项用于联机调试。

6）工具（Tools）　菜单项可以调用复杂指令（如 PID 指令、NETR/NETW 指令和 HSC 指令），安装文本显示器 TD200，改变用户界面风格（如设置按钮及按钮样式、添加菜单项），用“选项”子菜单可以设置三种程序编辑器的风格（如语言模式、颜色等）。

7）窗口（Windows）　菜单项的功能是打开一个或多个窗口，并进行窗口间的切换，还可以设置窗口的排放方式（如水平、垂直或层叠）。

8）帮助（Help）　菜单项可以方便地检索各种帮助信息，还提供网上查询功能，而且在软件操作过程中，可随时按［F1］键来显示在线帮助。

2. 工具栏

工具栏是一种代替命令或下拉菜单的便利工具，如图 1.3-12 所示，将 STEP7-Micro/WIN V4.0 编程软件最常用的操作以按钮形式设定到工具栏中，以提供简便的鼠标操作。用户可以定制每个工具栏的内容和外观，还可以用鼠标拖动工具栏，放到用户认为合适的位置。



图 1. 3-12　工具栏

可以用“查看”菜单中的“工具栏”选项来显示或隐藏四种按钮：标准、调试、公用和指令。四种按钮分别如图 1. 3-13 ~ 图 1. 3-16 所示。

标准工具栏中的按钮依次是新建项目、打开项目、保存项目、打印、打印预览、剪切、复制、粘贴、撤销、编译、全部编译、上载、下载、升序排列、降序排列和选项。



图 1. 3-13　标准工具栏

调试工具栏中的按钮依次是运行、停止、程序状态监控、暂停程序状态监控、状态表监控、趋势图、暂停趋势图、单次读取、全部写入、强制、取消强制、取消全部强制和读取全部强制。



图 1. 3-14　调试工具栏

公用工具栏中的按钮依次是插入网络、删除网络、切换 POU 注释、切换网络注释、切换符号信息表、切换书签、下一个书签、上一个书签、清除全部书签、应用项目中的所带符号和建立未定义符号表。

指令工具栏中的按钮依次是向下连线、向上连线、向左连线、向右连线、触点、线圈和指令盒。



图 1. 3-15　公用工具栏



图 1. 3-16　指令工具栏

3. 浏览条

在编程过程中，浏览条提供窗口快速切换的功能，可用主菜单中的“查看”→“框架”→“浏览条”选项控制是否打开浏览条。浏览条中包括“查看”和“工具”两个组件框，其中“查看”组件框含有以下八种组件。

1）程序块（Program Block）　由可执行的程序代码和注释组成。可执行的程序代码由主程序、可选的子程序和中断程序组成。代码被编译并下载到 PLC 中时，程序注释被忽略。S7-200 工程项目中规定的主程序只有一个，用 MAIN（OB1）表示；子程序有 64 个，用 SBR0 ~ SBR63 表示；中断程序有 128 个，用 INT0 ~ INT127 表示。

2）符号表（Symbol Table）　用来建立自定义符号与绝对地址间的对应关系，并可附加注释，使得用户可以使用具有实际含义的符号作为编程元件，增加程序的可读性。当程序编译后下载到 PLC 中时，所有的自定义符号都将被转换成绝对地址，而自定义符号被忽略。

3）状态表（Status Chart） 用于联机调试时监视指定的内部变量的状态和当前值。状态表并不下载到 PLC，仅仅是监控用户程序运行情况的一种工具。监控用户程序运行时，只需要在地址栏中写入变量地址，在数据格式栏中标明变量的类型，就可以监视这些变量的状态和当前值。

4）数据块（Data Block） 由数据（存储器的初始值和常数值）和注释组成，可以对变量寄存器 V 进行初始数据的赋值或修改，并可附加必要的注释。数据被编译并下载到 PLC，注释被忽略。对于继电器—接触器控制系统的数字量控制一般只有主程序，不使用子程序、中断程序和数据块。

5）系统块（System Block） 主要用于系统组态。系统组态主要包括设置数字量或模拟量输入滤波、设置脉冲捕捉、配置输出表、定义存储器保持范围、设置密码和通信参数等。

6）交叉引用（Cross Reference） 可以列举出程序中使用的各操作数在哪一个程序块的什么位置出现，以及使用它们的指令助记符；可以查看哪些内存区域已经被使用，作为位使用还是作为字节使用等；在运行方式下编辑程序时，还可以查看程序当前正在使用的跳变信号的地址。交叉引用表不能下载到 PLC，程序编译成功后才能看到交叉引用表的内容。在交叉引用表中双击某个操作数时，可以显示含有该操作数的那段程序。

7）通信（Communications） 可用来建立计算机与 PLC 之间的通信连接，以及设置和修改通信参数。

在浏览条中单击“通信”图标，则会出现一个“通信”对话框，双击其中的“PC/PPI 电缆”图标，将出现“PG/PC 接口”对话框，此时可以安装或删除通信接口，检查各参数设置是否正确，其中波特率的默认值是 9.6kbit/s。

设置好参数后，就可以建立与 PLC 的通信联系了。双击“通信”对话框中的“刷新”图标，STEP7-Micro/WIN V4.0 将检查所有已连接的 S7-200 的 CPU 站，并为每一个站建立一个 CPU 图标。

建立计算机与 PLC 的通信联系后，可以设置 PLC 的通信参数。首先单击浏览条中的“系统块”图标，将出现“系统块”对话框；然后单击“通信端口（PORT）”选项，检查和修改各参数，确认无误后，单击“确认（OK）”按钮；最后单击工具栏的“下载（Download）”按钮，即可把确认后的参数下载到 PLC 主机。

8）设置“PG/PC”接口 单击浏览条中的“PG/PC 接口”按钮，再单击“设置 PG/PC 接口”对话框中的“属性”按钮，可以为 STEP7-Micro/WIN V4.0 选择网络地址和波特率。

“工具”组件框包括指令向导、文本显示向导、位置控制向导、EM253 控制面板、调制解调器扩展向导、以太网向导、AS-i 向导、因特网向导、配方向导、数据记录向导、PID 调节控制面板、S7-200 EIplorer 和 TD KeQpad Designer。

4. 指令树

指令树提供编程所用到的所有命令和 PLC 指令的快捷操作，可以用主菜单中的“查看”→“框架”→“指令树”选项控制是否打开指令树。

5. 输出窗口

输出窗口用来显示程序编译的结果信息，如各程序块（主程序、子程序数量及子程序号、中断程序数量及中断程序号等）及各块大小、编译结果有无错误以及错误编码及其位置。输出窗口可用主菜单中的“查看”→“框架”→“输出窗口”选项控制其是否打开。

6. 状态栏

状态栏也称任务栏，用来显示软件执行情况，编辑程序时显示光标所在的网络号、行号和列号，运行程序时显示运行的状态、通信波特率、远程地址等信息。

7. 程序编辑器

用户可以用梯形图、语句表或功能块图程序编辑器编写和修改用户程序。程序编辑器包含局部变量表和程序视图（梯形图、语句表和功能块图）窗口。如果需要，用户可以拖动分割条，扩展程序视图，并覆盖局部变量表。当用户在主程序之外建立子程序或中断程序时，标记出现在程序编辑器窗口的底部，可单击该标记，在子程序、中断程序和主程序之间移动。

每个程序块都对应一个局部变量表，在带参数的子程序调用中，参数的传递就是通过局部变量表进行的。局部变量表包含对局部变量所做的赋值（子程序和中断程序使用的变量）。

四、计算机与 PLC 的通信连接

1. 开发 S7-200 PLC 所需要的硬件

开发 S7-200 PLC 需要用户有一台装有 STEP 7-Micro/WIN V4.0 编程软件的计算机、S7-200 PLC和一根 PC/PPI 编程电缆（见图 1.3-17）。

2. 硬件连接

图 1.3-18 为一个常见的硬件连接图，连接步骤如下：

1）将 PC/PPI 多主站电缆 RS232 端（标识为“PC”）连接到计算机的串行通信端口 COM1（COM2 亦可）。

2）将 PC/PPI 多主站电缆 RS485 端（标识为“PPI”）连接到 S7-200 CPU 模块的通信端口 PORT0 或 PORT1。

3）如图 1.3-19 所示，设置 PC/PPI 多主站电缆的 DIP 开关。



图 1.3-17　PC/PPI 编程电缆

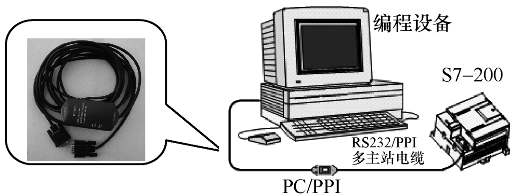


图 1.3-18　PLC 与 PC 的硬件连接

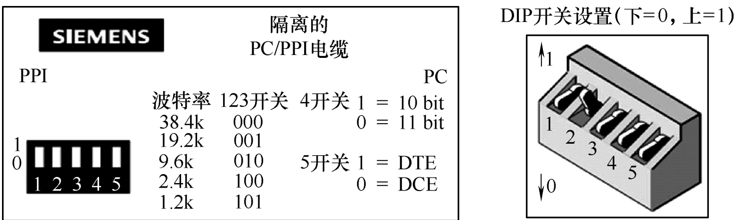


图 1.3-19　PC/PPI 多主站电缆 DIP 开关的设置

PC/PPI 多主站电缆中间有通信模块，模块外部设有波特率设置开关，有五种支持 PPI 协议的波特率可以选择，分别是 1.2kbit/s、2.4kbit/s、9.6kbit/s、19.2kbit/s 和 38.4kbit/s，系统的默认值为 9.6kbit/s。在用 PC/PPI 电缆上的 DIP 开关设置波特率时，应与编程软件中设置的波特率相同。

DIP 开关上有 5 个开关键，1、2、3 号键用于设置波特率，通信速率的默认值为 9.6kbit/s，则 1、2、3 号键设置为 010。DIP 开关的第 4 位用于选择 10 位或 11 位通信模式，第 5 位用于选择将 RS232 口设置为数据终端设备（DTE）模式或数据通信设备（DCE）模式。未使用调制解调器时，4、5 号键均应设置为 0。

3. 设置通信参数

在使用 PC/PPI 多主站电缆完成计算机与 PLC 之间的硬件连接后，按照以下步骤设置通信参数。

1）将 PLC 前盖内的模式选择开关拨至“STOP”。

2）合上断路器 QF，给 PLC 上电，这时“STOP”状态指示灯黄灯亮。

3）在 STEP7-Micro/WIN V4.0 运行时单击浏览条中的“通信”图标，或从主菜单“查看”→“组件”中选择“通信”选项，则会出现一个“通信”对话框，如图 1.3-20 所示。

在图 1.3-20 所示的“通信”对话框内右侧显示编程计算机将通过 PC/PPI 电缆尝试与 CPU 通信，并且本地编程计算机的网络通信地址是“0”。

4）双击“通信”对话框中的 PC/PPI 电缆图标，将出现“设置 PG/PC 接口”对话框，如图 1.3-21 所示。



图 1.3-20 “通信”对话框

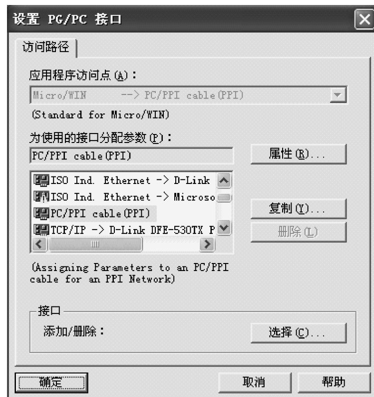


图 1.3-21 “设置 PG/PC 接口”对话框

5）单击“属性”按钮，将出现“属性—PC/PPI cable（PPI）”对话框，如图 1.3-22 所示，在“本地连接”选项卡中，选择连接到计算机的串行通信端口 COM1。

在“PPI”选项卡中，单击“默认”按钮，可获得默认的参数，如图 1.3-23 所示。

在站参数（Station Parameters）的“地址（Address）”微调框中，运行 STEP7-Micro/WIN V4.0 的计算机（主站）的默认站地址为 0。在“超时（Timeout）”下拉列表框中设置建立通信的最长时间，默认值为 1s。

在网络参数中，“高级 PPI（Advanced PPI）”的功能是允许在 PPI 网络中与一个或多个

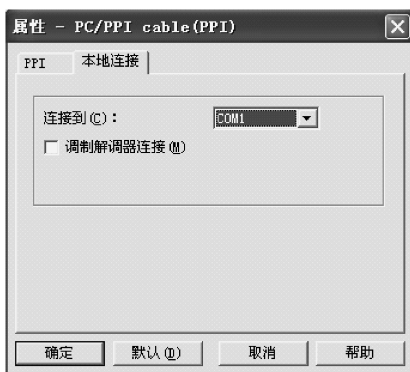


图 1.3-22　“属性—PC/PPI cable（PPI）”对话框中的“本地连接”选项卡



图 1.3-23　“属性—PC/PPI cable（PPI）”对话框中的“PPI”选项卡

S7-200 CPU 建立连接。S7-200 CPU 的通信口 0 和通信口 1 分别可以建立 4 个连接。选择“多主站网络（Multiple Master Network）”，即可以启动多主站模式，未选时为单主站模式。在多主站模式中，编程计算机和 HMI（如 TD200 和触摸屏）是通信网络中的主站，S7-200 CPU 是从站。在单主站模式中，用于编程的计算机是主站，一个或多个 S7-200 CPU 是从站。使用了多主站 PPI 电缆，可以忽略“多主站网络”和“高级 PPI”复选框。传输率的默认值为 9.6kbit/s。根据网络中的设备数选择最高站地址，默认值为 31。

以上默认参数一般不必改动，核实之后直接单击“确定”按钮，回到图 1.3-20 所示的“通信”对话框。

4. 建立计算机与 PLC 在线联系

在“通信”对话框中双击“双击刷新”图标，将检查所连接的所有 S7-200 CPU 站，并为每个站建立一个 CPU 图标，并显示该 CPU 的型号、版本号和网络地址，如图 1.3-24 所示，即建立起计算机与 PLC 在线联系。

关闭“通信”对话框后，可以发现指令树项目 1 显示实际连接并通信成功的 CPU 型号和版本信息，如图 1.3-25 所示。

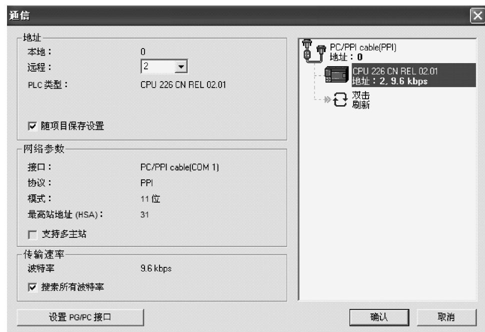


图 1.3-24　搜索结果

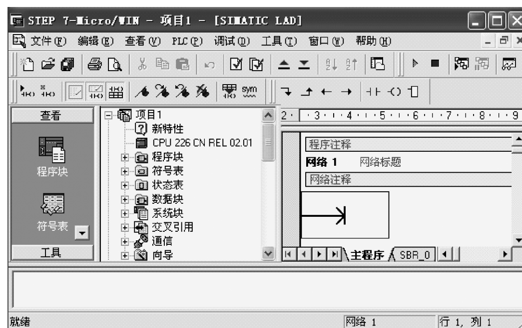


图 1.3-25　指令树项目 1 显示的 CPU 型号和版本信息

5. 检查、设置和修改 PLC 的通信参数

计算机与 PLC 建立起在线连接后，即可以利用软件检查、设置和修改 PLC 的通信参数，

步骤如下：

1）单击浏览条中的“系统块”图标，或从主菜单“查看”→“组件”中选择“系统块”选项，将出现“系统块”对话框，如图 1.3-26 所示。

2）单击“通信端口”选项卡，检查各参数，确认无误后单击“确认”按钮。若要修改某些参数，可以先进行有关的修改，再单击“确认”按钮。

6. 读取 PLC 的信息

利用菜单命令“PLC”→“信息”，可读取 PLC 的信息，如 PLC 的运行状态、扫描速率及 CPU 的型号等信息，如图 1.3-27 所示。

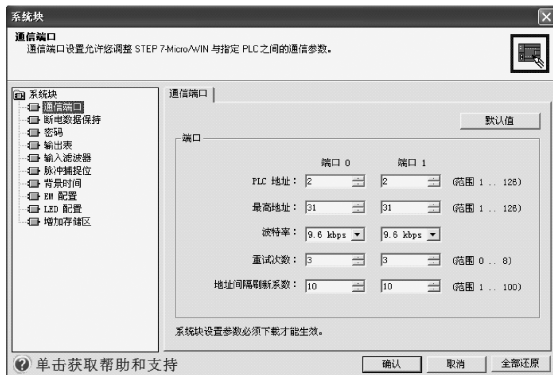


图 1.3-26 “系统块”对话框

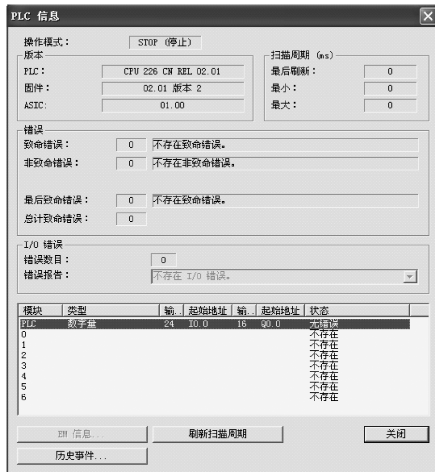


图 1.3-27 显示 PLC 信息

五、一个简单程序的编辑与调试运行

1. 创建新项目

在桌面上直接双击 STEP7-Micro/WIN V4.0 编程软件的快捷图标，自动创建一个新的工程项目——项目 1。

2. 选择指令集和编辑器

S7-200 PLC 支持的指令集有 SIMATIC 和 IEC1131-3 两种。SIMATIC 是专为 S7-200 PLC 设计的，专用性强。采用 SIMATIC 指令编写的程序执行时间短，可以使用 LAD、STL、FBD 三种编辑器。

选择菜单命令“工具”→“选项”，在选项树中选择“常规”。在“常规”选项卡中，“默认编辑器”选择“梯形图编辑器”，“编程模式”选择“SIMATIC”，“语言”选择“中文”，如图 1.3-28 所示，单击“确认”按钮。

3. 选择 PLC 类型

方法 1：选择菜单命令“PLC”→“类型”。

方法 2：在指令树中选择双击项目 1 名称下的默认类型 CPU221 REL 02.01。

在弹出的“PLC 类型”对话框中，“PLC 类型”下拉列表框里选择 CPU226，如图 1.3-29 所示，单击“确认”按钮。



图 1.3-28　指令集和编辑器选择



图 1.3-29　PLC 类型选择

4. 保存项目

选择菜单命令“文件”→“保存”或“另存为”或单击工具栏的“保存项目”按钮，在弹出的“另存为”窗口中可以选择默认路径“c：\Program Files\Siemens\STEP 7-MicroWIN V4.0\Projects”，文件名输入为“三个单开关控制两只灯”，然后单击“保存”按钮，如图 1.3-30 所示。

5. 编辑符号表

单击浏览条中的“符号表”图标，打开符号表编辑器，在符号表中输入符号及相应地址，也可以输入注释，如图 1.3-31 所示。

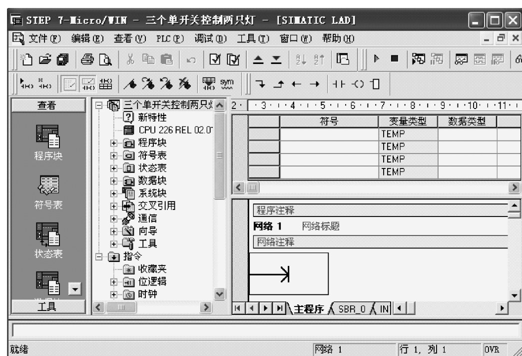


图 1.3-30　三个单开关控制两只灯项目



图 1.3-31　编辑符号表

6. 输入梯形图程序

单击浏览条中的“程序块”图标，打开程序编辑器，按以下步骤输入如图 1.3-32 所示的三个单开关控制两只灯的梯形图程序。

（1）输入常开触点 I0.0

1）利用指令树按钮，输入步骤如图 1.3-33 所示。

① 在指令树的“指令”选项下单击打开所需要的指令类别，如单击“位逻辑”左面的 ⊞，或双击 **位逻辑**，如图 1.3-33①所示；

② 从打开的“位逻辑”指令树中选择需要的常开触点元件 -| |-，如图 1.3-33②所示；

③ 按住鼠标左键拖动所选择的常开触点元件 -| |- 到程序编辑器窗口中所需要的位置，如

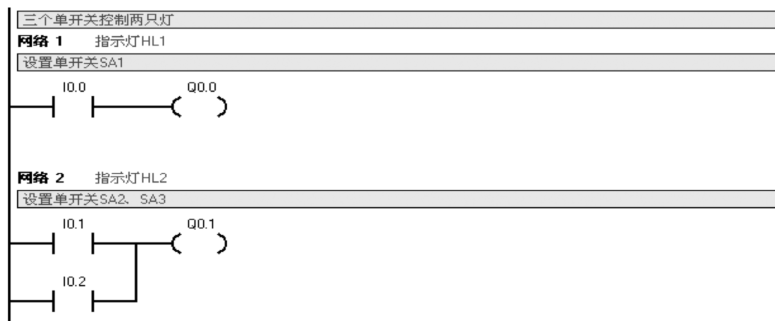


图 1.3-32　三个单开关控制两只灯梯形图

图 1.3-33③所示；

④ 释放鼠标左键，常开触点元件 -| |- 就放置在所需要的位置了，如图 1.3-33④所示；

⑤ 单击“?? . ?”处，如图 1.3-33⑤所示；

⑥ 在“?? . ?”处输入常开触点元件的地址，如 I0.0，如图 1.3-33⑥所示；

⑦ 按回车键确认，光标自动右移一格，如图 1.3-33⑦所示，一个指令就输入完毕了。

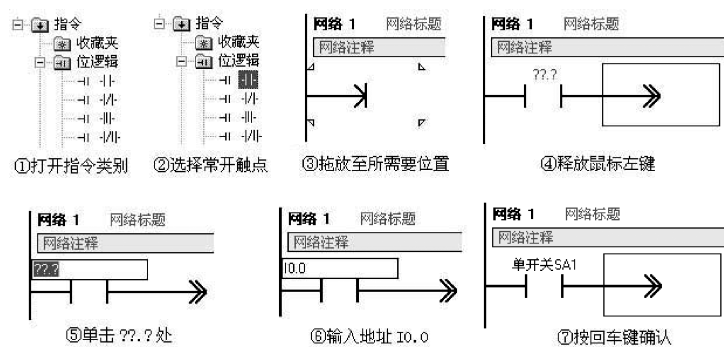


图 1.3-33　利用指令树按钮输入指令

除了利用指令树按钮输入指令外，还可以利用工具栏按钮输入指令。例如，单击工具栏上的指令触点、线圈或指令盒按钮，会分别出现一个下拉列表，如图 1.3-34 所示。

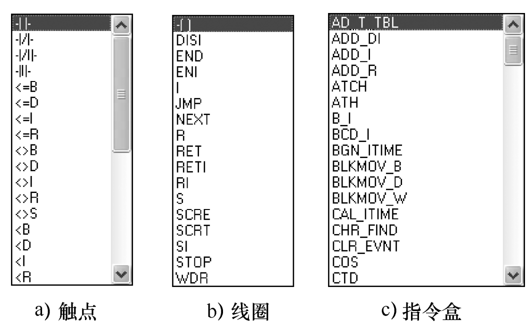


图 1.3-34　触点、线圈或指令盒指令列表

2）利用工具栏按钮，输入步骤如图 1.3-35 所示。

① 在程序编辑窗口中将光标定位到所要编辑的位置，如图 1.3-35①所示；

② 单击工具栏上的触点指令，出现一个下拉列表，如图 1.3-35②所示；

③ 利用滚动条或键盘的↑、↓键浏览至所需的指令，如 -| |-，单击 -| |- 指令或使用回车键即可输入该指令到所要编辑的位置，如图 1.3-35③所示；

④ 单击“?? . ?”处，如图 1.3-35④所示；

⑤ 在“?? . ?”处输入常开触点元件的地址，如 I0. 0，如图 1. 3-35⑤所示；

⑥ 按回车键确认，光标自动右移一格，如图 1. 3-35⑥所示，一个指令就输入完毕了。

（2）输入线圈 Q0. 0　把光标放在常开触点 I0. 0（单开关 SA1）的后面一格位置，即图 1. 3-35 第⑥步的光标所放位置，采用与输入常开触点 I0. 0 一样的方法输入线圈 Q0. 0，只是编程元件选择的是线圈 -[ ]，物理地址是 Q0. 0，符号地址是指示灯 HL1。输入完线圈 Q0. 0，网络 1 输入完毕。

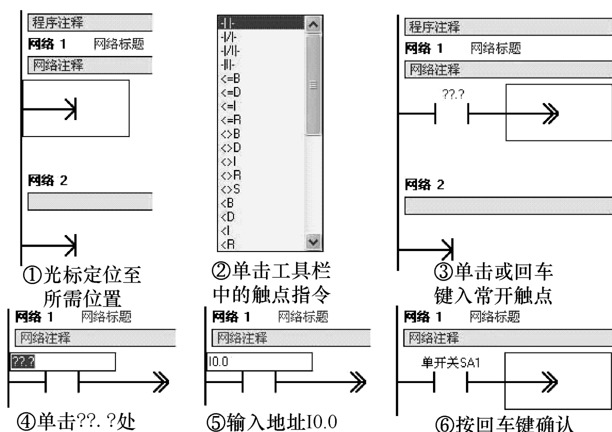


图 1. 3-35　利用工具栏按钮输入指令

（3）输入常开触点 I0. 1　把光标放在网络 2 的程序段起始编辑位置，采用与输入常开触点 I0. 0 一样的方法输入常开触点 I0. 1，只是常开触点的物理地址是 I0. 1，符号地址是单开关 SA2。

（4）输入常开触点 I0. 2　步骤如图 1. 3-36所示。

1）把光标放在图 1. 3-36①所示的位置；

2）输入常开触点 I0. 2，如图 1. 3-36②所示；

3）把光标放在常开触点 I0. 2（单开关 SA3）上，如图 1. 3-36③所示；

4）单击工具栏的向上连线按钮，如图 1. 3-36④所示，常开触点 I0. 2（单开关 SA3）与常开触点 I0. 1（单开关 SA2）并联上了。

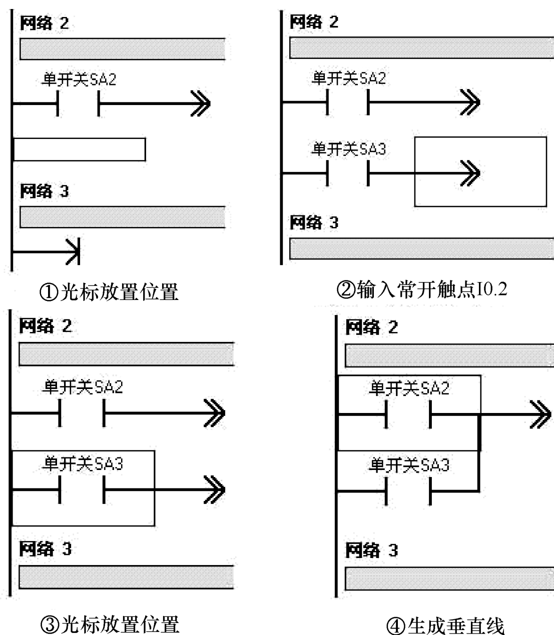


图 1. 3-36　输入常开触点 I0. 2 的步骤

（5）输入线圈 Q0. 1　把光标放在常开触点 I0. 1（单开关 SA2）的后面一格位置，采用与输入线圈 Q0. 0 一样的方法输入线圈 Q0. 1，只是线圈的物理地址是 Q0. 1，符号地址是指示灯 HL2。

最后填写上程序注释、网络标题及网络注释，梯形图输入完毕。在 SIMATIC LAD 窗口中，如果只显示元件的绝对地址，只要选择菜单命令“查看”→“符号寻址”，则“符号寻址”前面的“√”去掉，即可得到梯形图程序。

当梯形图输入完毕后，要进行程序的保存。单击工具栏中的“保存项目”按钮，程序便保存下来了。

7. 编译程序

在 STEP7-Micro/WIN V4.0 中，打开所保存的“三个单开关控制两只灯”梯形图，单击工具栏中的“全部编译”按钮，或者选择菜单命令“PLC”→“全部编译”，程序便被编译成 PLC 能够识别的机器码。

8. 连接 PLC 与计算机

根据上述介绍的方法进行硬件连接。

9. 下载程序

选择菜单命令“文件”→“下载”，或单击工具栏中的“下载”按钮，弹出如图 1.3-37所示的“下载”对话框。

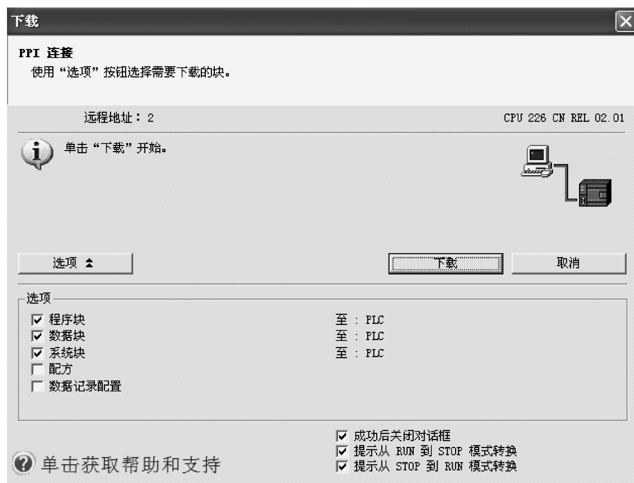


图 1.3-37　“下载”对话框

图 1.3-37 中，通常在“选项”区域中选择“程序块”、“数据块”和“系统块”三个选项，再单击“下载”对话框中的“下载”按钮即可。图 1.3-38 所示为正在下载程序的界面。



图 1.3-38　下载进行中的界面

10. 运行

将 CPU226 前盖内的模式选择开关拨至“TERM”，选择菜单命令“PLC”→“RUN（运行）”，或单击工具栏中的“运行”按钮，自动弹出是否运行的对话框，如图 1.3-39 所示。确认运行后单击“是”按钮，CPU 开始运行用户程序，CPU 上的 RUN 指示灯绿灯亮，STOP 指示灯黄灯灭。



图 1.3-39　是否运行对话框

11. 监控

（1）程序状态监控

1）选择菜单命令“调试”→“开始程序状态监控”或单击工具栏上的按钮，则程序状态监控初始画面如图 1.3-40 所示。

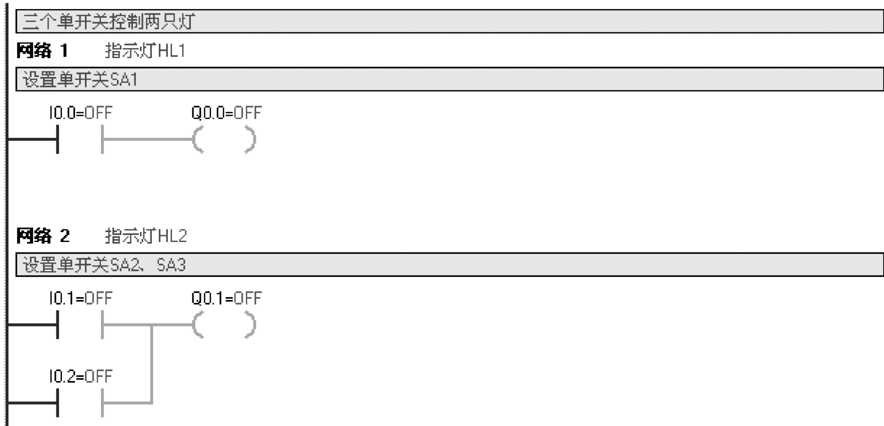


图 1.3-40　程序状态监控初始画面

2）分别依次操作单开关 SA1、SA2、SA3，观察并记录实际运行情况及程序状态监控画面相应变化情况。如果运行正确，则实际运行情况和监控画面变化情况应该是一致的。

3）选择菜单命令“调试”→“停止程序状态监控”或再次单击工具栏上的按钮，则停止程序状态监控。

（2）状态表监控元件状态

1）单击浏览条中的“状态表”图标或选择“查看”→“组件”→“状态表”菜单命令，打开状态表画面，并在状态表画面的地址列输入所需要监控的元件和选择格式，如图 1.3-41 所示。



	地址	格式	当前值	新值
1	IB0	十六进制		
2	QB0	十六进制		
3		有符号		
4		有符号		
5		有符号		

图 1.3-41　状态表

2）选择菜单命令“调试”→“开始状态表监控”或单击工具栏上的“状态表监控”按钮，则状态表监控初始画面如图 1.3-42 所示。



图 1.3-42　状态表监控初始画面

3）分别依次操作单开关 SA1、SA2、SA3，观察实际运行情况及状态表监控的元件状态相应变化情况。如果运行正确，则实际运行情况和状态表中监控的元件状态变化情况应该是一致的。

4）选择菜单命令“调试”→“停止状态表监控”或单击工具栏上的“状态表监控”按钮，则停止状态表监控。

12. 停止运行

如果停止运行用户程序，则选择菜单命令“PLC”→“STOP（停止）”，或单击工具栏中的“停止”按钮，自动弹出是否停止运行的对话框，如图 1.3-43 所示。确认停止运行后单击“是”按钮，CPU 停止运行用户程序，CPU 上的 STOP 指示灯黄灯亮，RUN 指示灯绿灯灭。



图 1.3-43　停止运行对话框

第四节　PLC 常用外部设备与接线

一、PLC 的输入设备与接线

PLC 输入端用来接收和采集用户输入设备产生的信号。这些输入设备主要有两种类型：一类是按钮、转换开关、行程开关、接近开关、光电开关、数字拨码开关与继电器触点等开关量输入设备；另一类是电位器、编码器和各种变送器等模拟量输入设备。正确地理解和连接输入与输出电路，是保证 PLC 安全可靠工作的前提。

1. 按钮、转换开关（见图 1.4-1）

利用按钮推动传动机构使动触点与静触点接通或断开并实现电路换接的开关是一种结构简单、应用十分广泛的主令电器，在电气自动控制电路中，用于手动发出控制信号，给 PLC 输入端子输送输入信号。按钮、转换开关与 PLC 输入端子的接线如图 1.4-2 所示。



图 1. 4-1　按钮、转换开关实物图

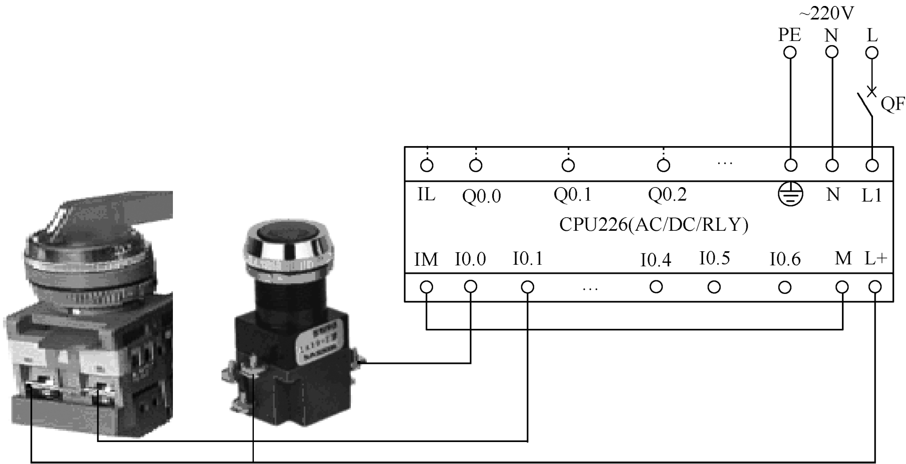


图 1. 4-2　按钮、转换开关与 PLC 输入端子的接线示意图

2. 行程开关、接近开关、光电开关（见图 1. 4-3）



图 1. 4-3　行程、接近与光电开关实物图

（1）行程开关　行程开关是利用生产机械运动部件的碰压，使其触头动作，从而将机械信号转变为电信号，使运动机械按一定的位置或行程实现自动停止、反向运动、变速运动或自动往返运动。行程开关与 PLC 输入端子的接线如图 1. 4-4 所示。

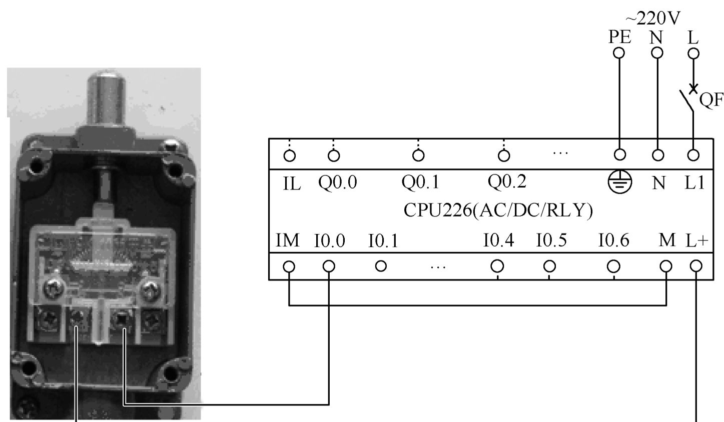


图 1.4-4　行程开关与 PLC 输入端子的接线示意图

（2）接近开关　接近开关可以在不与目标物实际接触的情况下检测靠近开关的金属目标物。根据操作原理，接近开关大致可以分为电磁感应的高频振荡型、磁力型和电容变化的电容型三大类，接近开关有两线制和三线制之区别，其接线也就有两线制和三线制两种。

1）三线制接线。三线制信号输出有 PNP（输出高电平约 24V）和 NPN（输出低电平 0V）两种形式，其接线也分为 PNP 和 NPN 形式。

① PNP 常开型接线。PNP 接通时为高电平输出，即接通时黑色线输出高电平（通常为 24V）。图 1.4-5a 为 PNP 型三线开关原理图，接近开关引出的三根线，棕色线接电源正极，蓝色线接电源负极，黑色线为控制信号线。此为常开开关，当开关动作时黑色线和棕色线接通，此时负载两端加上直流电压而获电动作。

② NPN 常开型接线。NPN 接通时为低电平输出，即接通时黑色线输出低电平（通常为 0V）。图 1.4-5b 为 NPN 型接近开关原理图，此为常开开关，当开关动作时黑色和蓝色两线接通，此时负载两端加上直流电压而获电动作。

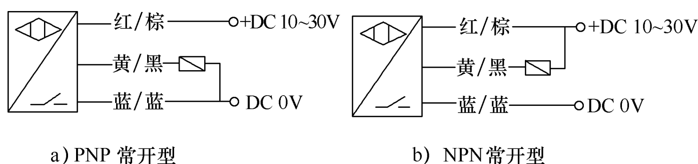


图 1.4-5　接近开关接线示意图

2）两线制接线。两线制接近开关的接线比较简单，接近开关与负载串联后接到电源，如图 1.4-6 所示。

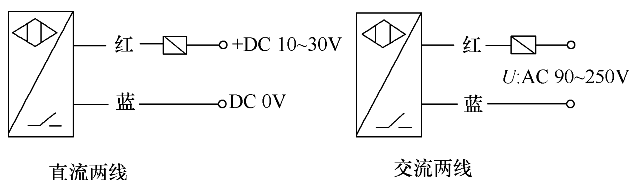


图 1.4-6　两线制接线示意图

（3）光电开关　光电开关是利用被检测物体对红外光束的遮光或反射，由同步回路选通而检测物体的有无，其物体不限于金属，对所有能反射光线的物体均可检测。光电开关与 PLC 的接线和接近开关与

PLC 的接线相同，图 1.4-7 为三线制的 PNP 型光电开关与 PLC 的接线示意图。PNP 型三线开关引出的三根线，棕色线接 PLC 传感器输出电源 24V 正极端子L+，蓝色线接 PLC 传感器输出电源负极端子 M，黑色线为控制信号线接 PLC 输入端子 I0.0。

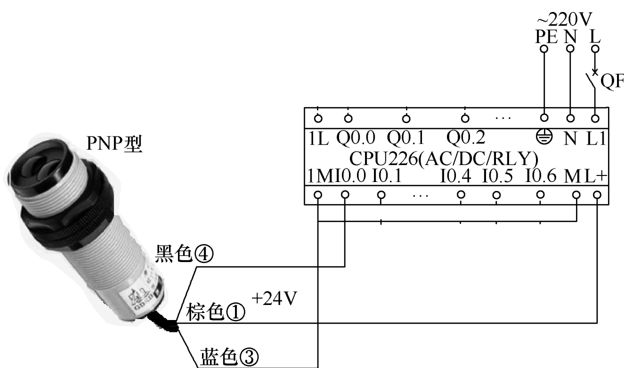


图 1.4-7 光电开关与 PLC 的接线示意图

3. 数字拨码开关

拨码开关在 PLC 控制系统中常常用到，图 1.4-8 为一位拨码开关的示意图。拨码开关有两种，一种是 BCD 码开关，即拨码数值从 0～9，输出为 8421 BCD 码；另一种是十六进制码，即从 0～F，输出为二进制码。拨码开关可以方便地进行数据变更，如控制系统中需要经常修改的数据，可使用拨码开关组成一组拨码器与 PLC 相接。图 1.4-9 所示为四位拨码开关与 PLC 输入接口的电路连接。四位拨码开关的 COM 端连在一起与 PLC 的 COM（公共）端相接，每位拨码开关的四条数据线按一定顺序接到 PLC 的四个输入点上。

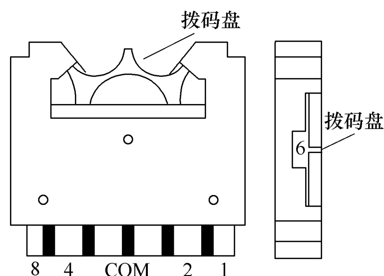


图 1.4-8 拨码开关示意图

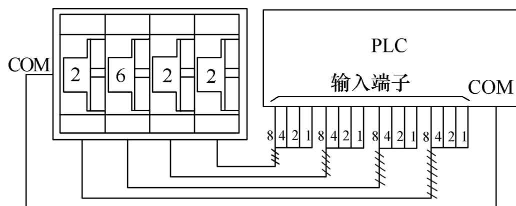


图 1.4-9 四位拨码开关与 PLC 的接线

4. 编码器

图 1.4-10 所示为常用的编码器，可将旋转编码器的输出脉冲信号直接输入给 PLC，利用 PLC 的高速计数器对其脉冲信号进行计数，以获得测量结果。不同型号的旋转编码器，其输出脉冲的相数也不同，有的旋转编码器输出 A、B、Z 三相脉冲，有的只有 A、B 相两相，最简单的只有 A 相。图 1.5-10 所示编码器有四条引线，其中两条是脉冲输出线，一条是 COM 端线，一条是电源线。编码器的电源可以是外接 DC 24V 电源，也可直接使用 PLC 的 DC 24V 电源。电源“－”端要与编码器的 COM 端连接，“＋”端与编码器的电源端连接。编码器的 COM 端与 PLC 输入 COM 端连接，A、B 两相脉冲输出线直接与 PLC 的输入端连接，A、B 为相差 90°的脉冲。旋转编码器还有一条屏蔽线，使用时要将屏蔽线接地，提高抗干扰性。图 1.4-11 为编码器与 PLC 的连接示意图，



图 1.4-10 编码器实物

A、B 两相分别与 PLC 的输入点（按高速计数器 HSC 的规定）进行连接。

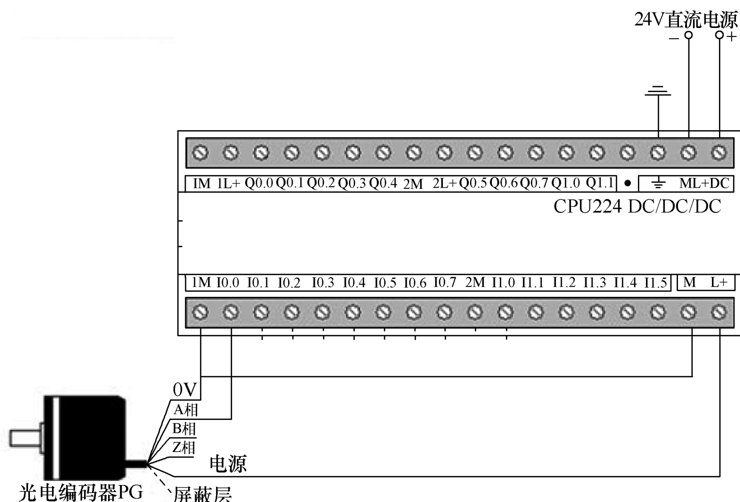


图 1.4-11　编码器与 PLC 的连接示意图

二、PLC 的输出设备与接线

PLC 的输出设备一般为接触器、指示灯、数码管、报警器、电磁阀、电磁铁、调节阀、调速装置等各种执行机构。正确地连接输出电路，是保证 PLC 安全可靠工作的前提，下面逐一介绍。

1. 接触器、微型继电器与 PLC 输出接线

接触器、微型继电器是一种自动的电磁式开关如图 1.4-12a 所示，当电磁线圈通入额定电压后，线圈电流产生磁场，使静铁心产生足够的吸力克服弹簧反作用力将动铁心向下吸合，常开触点闭合，常闭触点断开，通常应用于传统继电器控制电路和自动化控制电路中，在电路中起着自动调节、安全保护、转换电路等作用。图 1.4-12b 所示为继电器与 PLC 的输出接线，图中的电器元件线圈额定电压是交流 220V，如果是直流 24V，则需要外加直流 24V 的开关电源。接线时注意不同电压等级和性质的电源要独立接线，输出端子的公共端不能共用，如图 1.4-12c 所示的 1L 和 3L 公共端的接线。

2. 信号指示灯、声光报警器与 PLC 输出接线

在工业自动化控制系统中，为了安全和运行状况的指示，常常需要接入信号指示灯或声光报警器，如图 1.4-13 所示。信号指示灯、声光报警器与 PLC 的输出接线如图 1.4-14 所示，图中的电器元件额定电压为交流 220V。

3. 数码管与 PLC 输出接线

数码管可分为八段数码管和七段数码管，是一种半导体发光器件，其基本单元是发光二极管，八段数码管由八个发光二极管组成，七段数码管由七个发光二极管组成。通过对其不同的管脚输入相对的电流，使其发亮，可以显示十进制 0 ~ 9 的数字，也可以显示英文字母，包括十六进制中的英文 A ~ F。下面重点介绍七段数码管（见图 1.4-15）。七段数码管分为共阳极和共阴极如图 1.4-16 所示。共阳极七段数码管的正极（或阳极）为各段发光二极管

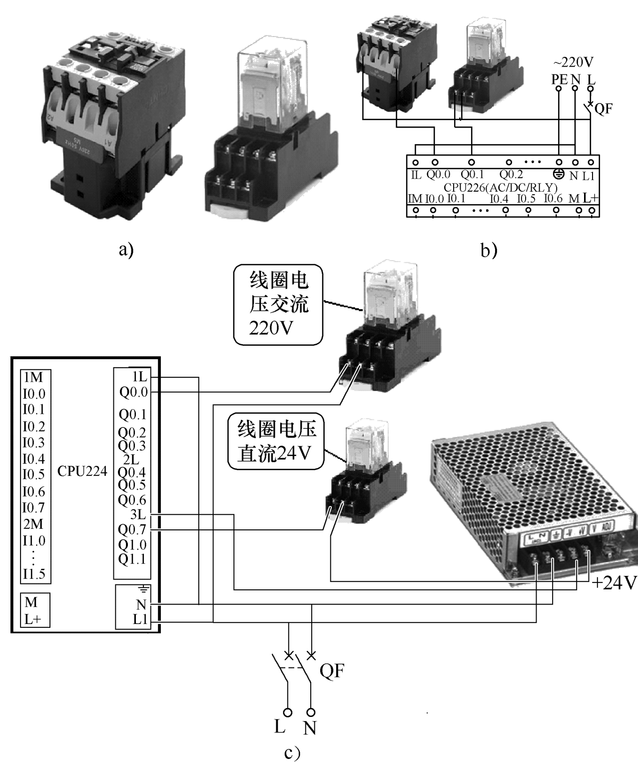


图 1. 4-12　继电器与 PLC 输出接线



图 1. 4-13　信号指示灯与声光报警器

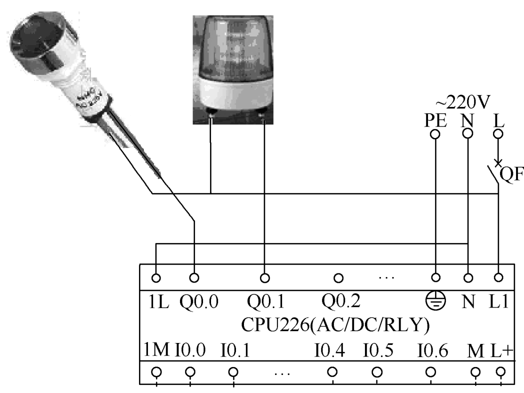


图 1. 4-14　信号指示灯、声光报警器与 PLC 的接线图

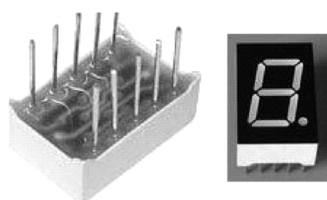


图 1. 4-15　数码管外形图

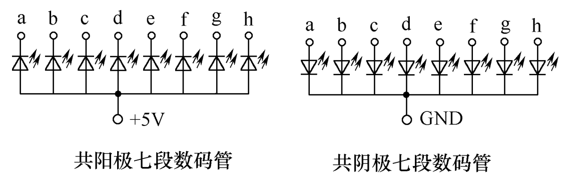


图 1. 4-16　七段数码管结构形式

的正极连接在一起，某段发光二极管的负极（或阴极）为低电平时，该段二极管发光。在共阴极结构中，各段发光二极管的阴极连在一起，将此公共点接地，某一段发光二极管的阳极为高电平时，该段二极管发光。共阴极七段数码管与 PLC 输出接线如图 1.4-17 所示。

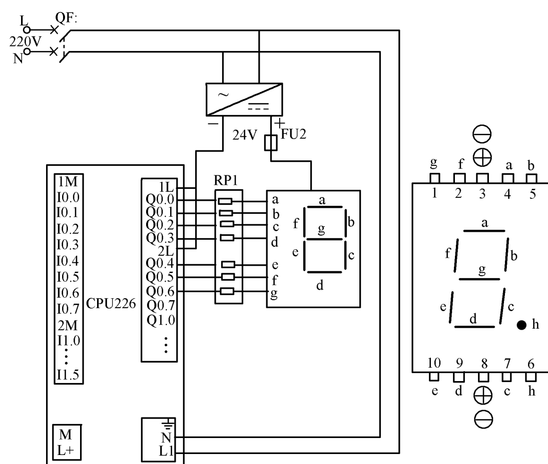


图 1.4-17　共阴极七段数码管与 PLC 输出接线图

第二章

S7-200 PLC 基本指令

S7-200 PLC 的基本指令中，位逻辑指令是最重要的，它是其他所有指令应用的基础。位逻辑指令在梯形图中是指对触点的简单连接和对标准线圈的输出，在语句表语言中是指对位存储单元的简单逻辑运算。位逻辑指令包括标准触点指令（LD、LDN、A、AN、O、ON）、输出指令（=）、置位和复位指令（S、R）、立即触点指令（LDI、LDNI、AI、ANI、OI、ONI）、立即输出指令（=I）、立即置位和立即复位指令（SI、RI）、逻辑堆栈指令（ALD、OLD、LPS、LRD、LPP、LDS）、上升沿检测指令和下降沿检测指令（EU、ED）、非指令（NOT）、置位优先双稳态触发器指令和复位优先双稳态触发器指令（SR、RS）以及空操作指令（NOP）。

一条指令由操作码和操作数组成。操作码表示要执行的功能，它告诉 CPU “做什么”；操作数则指定操作的对象，它以地址的方式表达，告诉 CPU 操作对象的存储位置。应该注意的是，有的指令是没有操作数的。

第一节　位逻辑指令

一、指令介绍

1. 触点类指令

在梯形图中常用的触点类指令见表 2. 1-1，主要进行触点的简单逻辑连接。

表 2. 1-1　触点类指令表

触　　点	梯形图符号	数据类型	操　作　数	
常开触点	bit ─┤ ├─	位	I、Q、V、M、SM、S、T、C、L	当常开触点对应存储器位（bit）为 1 时，表示该触点接通
常闭触点	bit ─┤/├─			当常开触点对应存储器位（bit）为 0 时，表示该触点接通
常开立即触点	bit ─┤I├─		I	当常开立即触点对应物理输入位（bit）为 1 时，表示该触点接通
常闭立即触点	bit ─┤/I├─			当常闭立即触点对应物理输入位（bit）为 0 时，表示该触点接通
上升沿检测触点	─┤P├─		I、Q、V、M、SM、S、T、C、L	上升沿触点检测到触点的每一次正跳变（从断开到接通瞬间）之后，触点就接通一次
下降沿检测触点	─┤N├─			下降沿触点检测到触点的每一次负跳变（从接通到断开瞬间）之后，触点就接通一次

2. 线圈类指令

在梯形图中常用的线圈类指令见表 2. 1-2，主要是对输出寄存器位的控制。

表 2. 1-2　线圈类指令表

线　圈	梯形图符号	数据类型	操作数	
线圈输出	bit —( )	位	I、Q、V、M、SM、S、T、C、L	当执行输出指令时，映像存储器指定的位（bit）被接通
线圈立即输出	bit —(I)		Q	当执行立即输出指令时，对应物理输出位（bit）被接通
线圈置位	bit —(S) N	位	I、Q、V、M、SM、S、T、C、L	当执行置位（置 1）指令时，从位（bit）指定的地址参数开始的 *N* 个点被置 1
线圈复位	bit —(R) N			当执行复位（置 0）指令时，从位（bit）指定的地址参数开始的 *N* 个点被置 0
线圈立即置位	bit —(SI) N		位：I、Q、V、M、SM、S、T、C、L N：VB、IB、QB、MB、SMB、SB、LB、AC、＊VD、＊AC、＊LD、常数	当执行立即置位（置 1）指令时，从位（bit）指定的地址参数开始的 *N* 个物理输出点被置 1
线圈立即复位	bit —(RI) N			当执行立即复位（置 0）指令时，从位（bit）指定的地址参数开始的 *N* 个物理输出点被置 0

二、指令练习

1. 触点指令练习

图 2. 1-1 所示为触点简单应用的一梯形图程序，图 2. 1-2 为梯形图对应的时序图。由图可知，当 I0. 0 和 I0. 1 都接通时，Q0. 0 接通，Q0. 2 保持原状态不变；当 I0. 1 断开时，Q0. 0 断开，在 I0. 1 断开的下降沿，触发 Q0. 2 接通一个扫描周期。

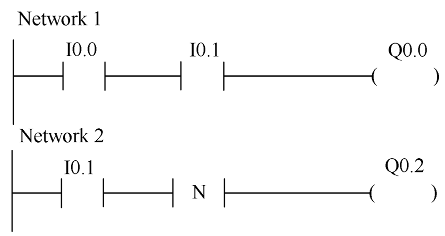


图 2. 1-1　梯形图

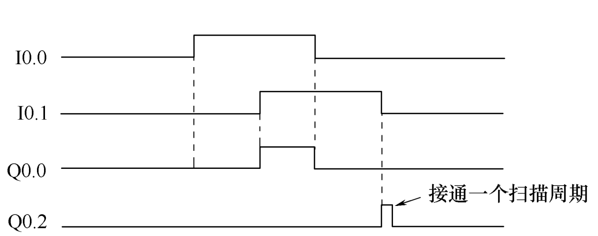


图 2. 1-2　时序图

2. 线圈指令练习

图 2. 1-3 所示为触点与线圈指令应用的梯形图程序，图 2. 1-4 为触点与线圈指令应用的梯形图对应的时序图。由图可知，当 I0. 0 接通时，Q0. 0 接通、Q0. 1 置 1 接通、Q0. 2 与

Q0. 3 这两位复位置 0。

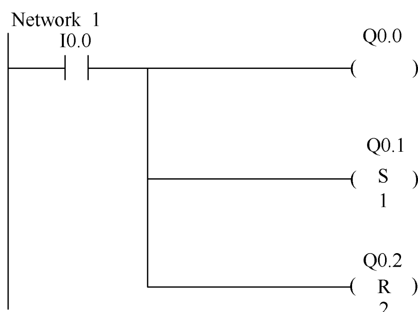


图 2. 1-3　触点与线圈指令应用的梯形图程序

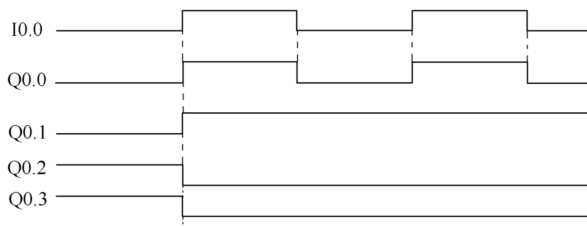


图 2. 1-4　触点与线圈指令应用的梯形图对应的时序图

三、梯形图的编程规则

尽管梯形图与继电器电路图在结构形式、元件符号及逻辑控制功能等方面相类似，但它们又有许多不同之处，梯形图具有自己的编程规则。

1）输入映像寄存器、输出映像寄存器、内部辅助继电器、定时器等元件的触点可多次重复使用，无须用复杂的程序结构来减少触点的使用次数。

2）梯形图的每一行都是从左边母线开始，线圈接在最右边或右母线（右母线可以不画出）上。触点不能放在线圈的右边，即线圈与右母线之间不能有任何触点，如图 2. 1-5 所示。

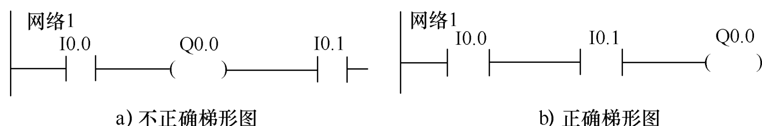


图 2. 1-5　线圈与触点的位置

3）线圈不能直接与左边母线相连，即左母线与线圈之间一定要有触点，如果需要，可以通过专用内部辅助继电器 SM0. 0 的常开触点连接，如图 2. 1-6 所示。SM0. 0 为 S7-200 PLC 中的常接通辅助继电器。

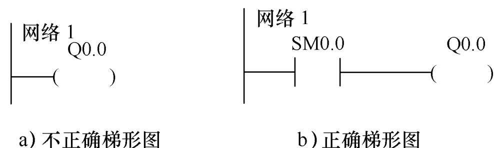


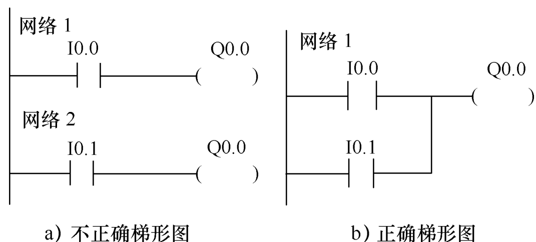
图 2. 1-6　SM0. 0 的应用

4）一般情况下，在梯形图中同一线圈只能出现一次。如果在程序中，同一线圈使用了两次或多次，称为“双线圈输出”。“双线圈输出”容易引起误操作，应避免线圈重复使用，如图 2. 1-7 所示。

5）梯形图必须符合顺序执行原则，即从左到右、从上到下地执行。不符合顺序执行的电路，不能直接编程，如图 2. 1-8 所示。

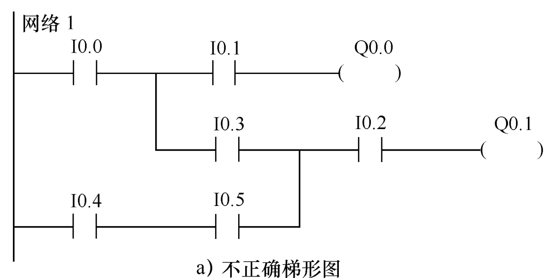
6）在梯形图中，有几个串联电路相并联时，应将串联触点多的回路放在上方；有几个并联电路相串联时，应将并联触点多的回路放在左方。这样所编制的程序简洁明了，指令条数减少，扫描周期缩短。图 2. 1-9 所示为梯形图程序的合理优化。

7）梯形图中的触点可以串联或并联，但继电器线圈只能并联而不能串联，如图 2. 1-10 所示。

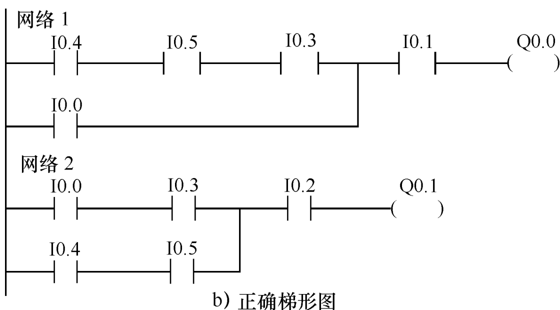


a）不正确梯形图　　b）正确梯形图

图 2. 1-7　相同编号的线圈程序

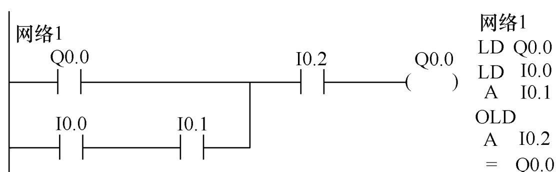


a）不正确梯形图

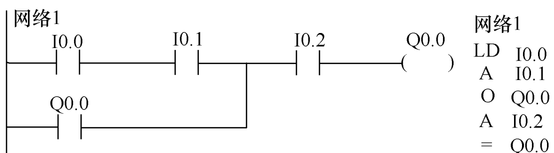


b）正确梯形图

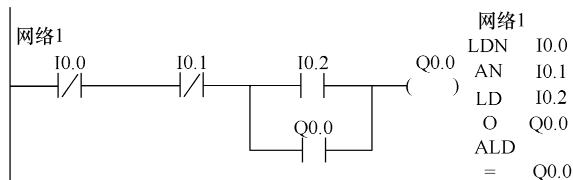
图 2. 1-8　不符合顺序执行编程规则的程序处理



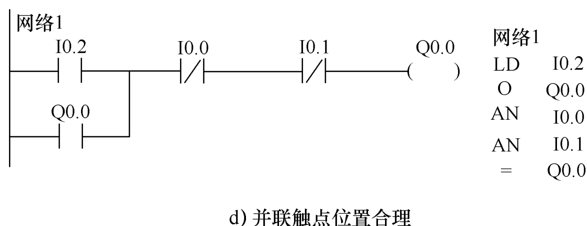
a）串联触点位置不当



b）串联触点位置合理

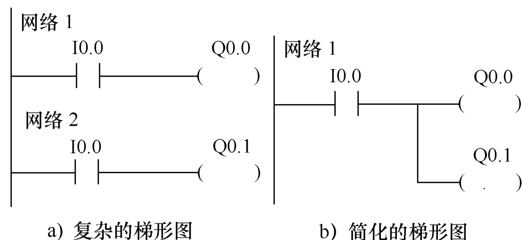


c）并联触点位置不当



d）并联触点位置合理

图 2. 1-9　合理优化的梯形图程序



a）复杂的梯形图　　b）简化的梯形图

图 2. 1-10　多线圈并联输出程序

第二节 定时器与计数器

一、指令介绍

1. 定时器

定时器是 PLC 中最常用的元件之一，用以实现时间的控制。在 S7-200 PLC 中的定时器按工作方式可分为接通延时定时器 TON、断开延时定时器 TOF 和保持型接通延时定时器 TONR 三种类型，按时基脉冲又可分为 1ms、10ms、100ms 三种，具体指令类别和定时器精度与编号见表 2.2-1 和表 2.2-2。

表 2.2-1 定时器的类别表

定时器类型	梯形图	指令	指令功能	数据类型及操作数
接通延时定时器	???? IN TON ????-PT	TON T＊＊＊，PT	使能端（IN）输入有效时，定时器开始计时，当前值从 0 开始递增，大于或等于预置值（PT）时，定时器位置 1（输出触点有效），当前值的最大值为 32 767。使能端无效（断开）时，定时器复位（当前值清零，输出状态位置 0）	T＊＊＊：字型，常数 T0～T255，指定时器编号 IN：位型，I、Q、V、M、SM、S、T、C、L、能流，指启动定时器 PT：整数型，IW、QW、VW、MW、SMW、T、C、LW、AC、AIW、＊VD、＊LD、＊AC、常数，指设定值输入端
断开延时定时器	???? IN TOF ????-PT	TOF T＊＊＊，PT	使能端（IN）输入有效时，定时器输出状态位置 1，当前值复位为 0。使能端（IN）断开时，开始计时，当前值从 0 递增，当前值达到预置值时，定时器状态位复位置 0，并停止计时，当前值保持	
保持型接通延时定时器	???? IN TONR ????-PT	TONR T＊＊＊，PT	使能端（IN）输入有效时，定时器开始计时，当前值递增，当前值大于或等于预置值（PT）时，输出状态位置 1。使能端输入无效时，当前值保持，使能端再次接通有效时，在原记忆值的基础上递增计时。保持型接通延时定时器采用线圈的复位指令（R）进行复位操作，当复位线圈有效时，定时器当前值清零，输出状态位置 0	

表 2.2-2 定时器的精度与编号

定时器类型	定时精度	定时范围及最大值	定时器号
TONR	1ms	定时时间 T = 时基 × 预置值；32.767s	T0，T64
	10ms	定时时间 T = 时基 × 预置值；327.67s	T1～T4，T65～T68
	100ms	定时时间 T = 时基 × 预置值；3276.7s	T5～T31，T69～T95
TON/TOF	1ms	定时时间 T = 时基 × 预置值；32.767s	T32，T96
	10ms	定时时间 T = 时基 × 预置值；327.67s	T33～T36，T97～T100
	100ms	定时时间 T = 时基 × 预置值；3276.7s	T37～T63，T101～T255

2. 计数器

计数器利用输入脉冲上升沿累计脉冲个数，在实际应用中用来对产品进行计数或完成复杂的逻辑控制任务。计数器的使用方法与定时器基本相似，编程时各输入端都应有控制信号，依据设定值及计数器类型决定动作时刻，以便完成计数控制任务。

S7-200 PLC 有递增计数（CTU）、增/减计数（CTUD）、递减计数（CTD）三类普通计数器，其编号为 C0 ~ C255，具体指令类别见表 2.2-3。

表 2.2-3 计数器指令类别表

计数器类型	梯形图	指令	指令功能	数据类型及操作数
递增计数 CTU	???? CU CTU R ????-PV	CTU C＊＊＊，PV	递增计数指令在 CU 端输入脉冲上升沿，计数器的当前值增 1 计数。当前值大于或等于预置值（PV）时，计数器状态位置 1，当前值累加的最大值为 32 767。复位输入（R）有效时，计数器状态位复位（置 0），当前计数值清零	C＊＊＊：字型，常数（0~255） CU、R：位，能流 PV：整数型，VW、IW、QW、MW、SMW、LW、AIW、AC、T、＊VD、＊AC、＊LD、＊SW、常数
递减计数 CTD	???? CD CTD LD ????-PV	CTD C＊＊＊，PV	复位输入（LD）有效时，计数器把预置值（PV）装入当前值存储器，计数器状态位复位（0）。CD 端每来一个输入脉冲上升沿，减计数器的当前值从预置值开始递减计数，当前值等于 0 时，计数器状态位置位（1），并停止计数	C＊＊＊：字型，常数（0~255） CD、LD：位，能流 PV：整数型，VW、IW、QW、MW、SMW、LW、AIW、AC、T、＊VD、＊AC、＊LD、＊SW、常数
增/减计数 CTUD	???? CU CTUD CD R ????-PV	CTUD C＊＊＊，PV	增/减计数器 CU 输入端用于递增计数，CD 输入端用于递减计数，指令执行时，CU/CD 端在计数脉冲的上升沿当前值增 1/减 1 计数。当前值大于或等于计数器预置值（PV）时，计数器状态位置 1。复位输入（R）有效或执行复位指令时，计数器状态位复位，当前值清零	C＊＊＊：字型，常数（0~255） CU、CD、R：位，能流 PV：整数型，VW、IW、QW、MW、SMW、LW、AIW、AC、T、＊VD、＊AC、＊LD、＊SW、常数

二、指令练习

1. 通电延时型定时器练习

图 2.2-1 为通电延时型定时器应用梯形图。当 I0.2 接通后，T33 定时器开始计时，当时间到达 3s 后 Q0.0 接通。程序状态时序图如图 2.2-2 所示，语句表如图 2.2-3 所示。

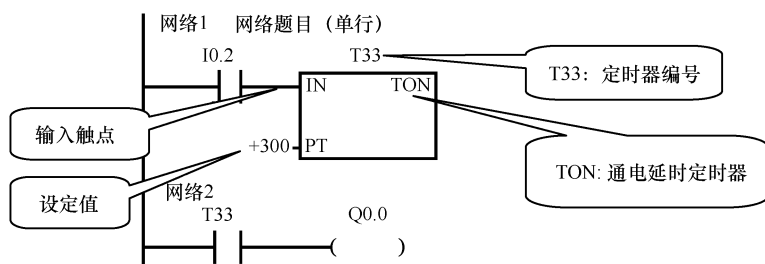


图 2. 2-1　通电延时型定时器应用梯形图

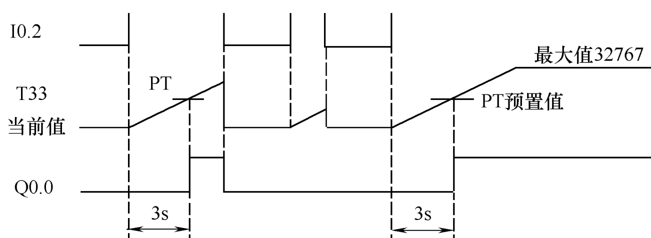


图 2. 2-2　时序图

```
NETWORK 1
LD      I0.2
TON     T33,+300
NETWORK 2
LD      T33
=       Q0.0
```

图 2. 2-3　语句表

使用说明：

1）使用 S7-200 PLC 的定时器，必须注意的是 1ms、10ms、100ms 定时器刷新方式不同。1ms 定时器由系统每隔 1ms 刷新一次，与扫描周期及程序处理无关，当扫描周期较长时，在一个扫描周期内多次被刷新，其当前值在每个扫描周期内可能不一致；10ms 定时器则由系统在每个扫描周期开始时自动刷新，因此每个周期只刷新一次，其当前值为常数；100ms 定时器则在该定时器指令执行时才被刷新。

2）因扫描方式不同，时基为 1ms 和 10ms 的定时器，一般不能用本身触点作为该定时器的激励输入条件；时基为 100ms 的定时器，用本身触点作为该定时器的激励输入条件时，定时器能正常工作。

3）一个定时器号不能同时用于 TOF 和 TON，例如，程序中不能有两个定时器 TON T32 和 TOF T32。

2. 断电延时型定时器练习

图 2. 2-4 为断电延时型定时器应用梯形图。当 I0. 0 接通时，Q0. 0 立即接通；当 I0. 0 断开后，定时器开始延时，延时时间 3s 后，Q0. 0 断开。程序状态时序图如图 2. 2-5 所示，语句表如图 2. 2-6 所示。

3. 递增计数器练习

图 2. 2-7 为递增计数器应用梯形图。当 I0. 0 触点由断开到接通时，CU 端接收一次脉冲，计数器的值加 1，当计数值大于或等于设定值 3 时，计数器 C5 的状态位被置 1，C5 触点接通，Q0. 0 接通；当复位（R）端的 I0. 1 接通时，C5 计数器复位，当前值清零，Q0. 0 断开。程序状态时序图如图 2. 2-8 所示，语句表如图 2. 2-9 所示。

使用说明：

1）程序中不能重复使用同一个计数器的编号，每个计数器只能使用一次。

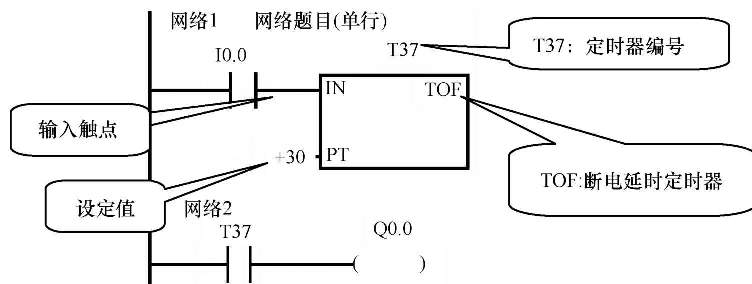


图 2. 2-4　断电延时型定时器应用梯形图

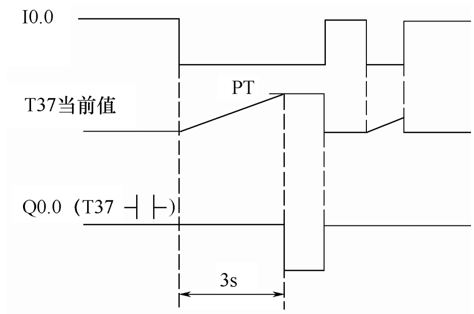


图 2. 2-5　时序图

```
NETWORK 1
LD      I0.0
TOF     T37, +30
NETWORK 2
LD      T37
=       Q0.0
```

图 2. 2-6　语句表

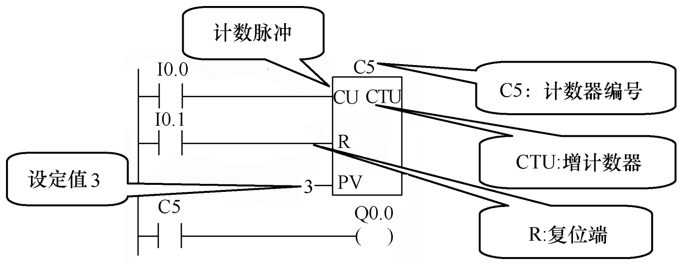


图 2. 2-7　递增计数器应用梯形图

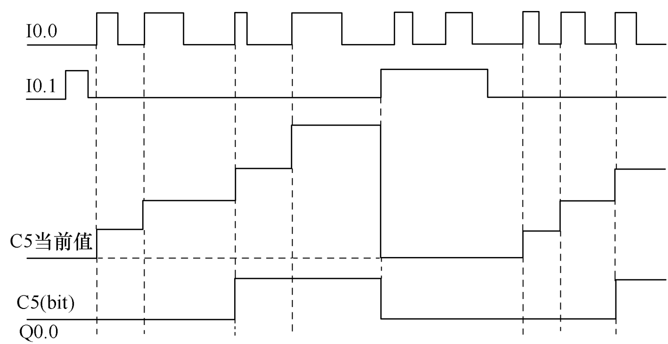


图 2. 2-8　时序图

```
LD    I0.0
LD    I0.1
CTU   C5,3
LD    C5
=     Q0.0
```

图 2. 2-9　语句表

2）对于增/减计数指令，当计数达到计数器最大值 32 767 后，下一个 CU 输入上升沿将使计数值变为最小值（－32 678）；同样达到最小值（－32 678）后，下一个 CD 输入上升

沿将使计数值变为最大值（32 767）。

3）因计数器不能自动复位，使用时要注意复位。

4. 递减计数器练习

图 2. 2-10 为递减计数器应用梯形图。当 I3. 0 触点由断开到接通时，CD 端接收一次脉冲，计数器的值减 1，当计数器的值减为 0 时，计数器 C50 的状态位被置 1，C50 触点接通，Q0. 0 接通；当复位（LD）端的 I1. 0 接通时，C50 计数器复位，当前值回复为设定值，Q0. 0 断开。程序状态时序图如图 2. 2-11 所示，语句表如图 2. 2-12 所示。

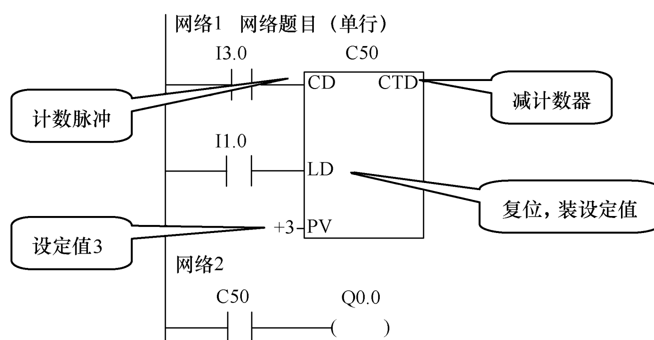


图 2. 2-10　递减计数器应用梯形图

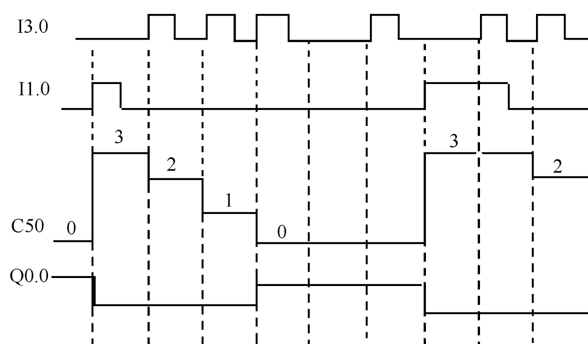


图 2. 2-11　时序图

```
NETWORK 1
LD      I3.0
LD      I1.0
CTD     C50, +3
NETWORK 2
LD      C50
=       Q0.0
```

图 2. 2-12　语句表

第三节　PLC 基本指令应用实例

实例 1　PLC 控制三相异步电动机连续运行

1. 传统继电控制线路

（1）控制要求

图 2. 3-1 所示为具有过载保护的接触器连续运行控制线路，其控制要求如下。

1）按下起动按钮，三相异步电动机单向连续运行。

2）按下停止按钮，三相异步电动机停止运行。

3）具有短路保护和过载保护等必要的保护措施。

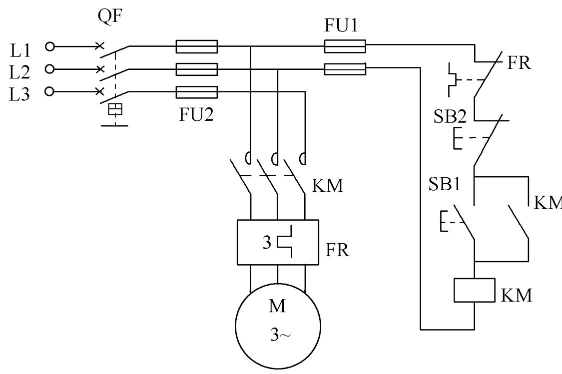


图 2. 3-1　具有过载保护的接触器连续运行控制线路

（2）工作原理分析

先合上电源开关 QF。

起动：按下按钮 SB1→SB1 常开触头接通→接触器 KM 线圈通电→

→接触器 KM 常开辅助触头接通（实现自保持）。

→接触器 KM（常开）主触头接通→电动机 M 通电起动并进入工作状态。

停止：按下按钮 SB2→SB2 常闭触头断开→接触器 KM 线圈断电→（解除自锁）KM（常开）主触头断开→电动机 M 断电并停止工作。

2. 用 PLC 控制电动机的连续运行

（1）I/O 地址通道分配

根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例起动按钮和停止按钮作为输入指令接在 PLC 的输入端子上，共 2 个输入点；接触器作为输出负载接在 PLC 的输出端子上，共 1 个输出点。I/O 的分配表见表 2. 3-1。

表 2. 3-1　I/O 的分配表

输入设备		输入继电器	输出设备		输出继电器
代号	功能		代号	功能	
SB1	起动按钮	I0. 0	KM1	接触器	Q0. 0
SB2	停止按钮	I0. 1			

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 2. 3-2）

1）用 PLC 改造电动机连续运行控制线路时，主电路不变，仅对控制电路进行改造。

2）设计电路原理图时，应具备完善的保护功能，PLC 外部硬件应具备过载保护电路。

3）PLC 继电器输出所驱动的负载额定电压一般不超过 220V，或设置外部中间继电器。

4）绘制原理图要完整规范。

（3）程序设计

参考程序 1：编程思路是采用起保停控制方式编写电动机连续运行控制梯形图程序，如图 2. 3-3 所示。

参考程序 2：编程思路是采用基本的置位与复位指令来编写电动机连续运行控制梯形图程序，如图 2. 3-4 所示。

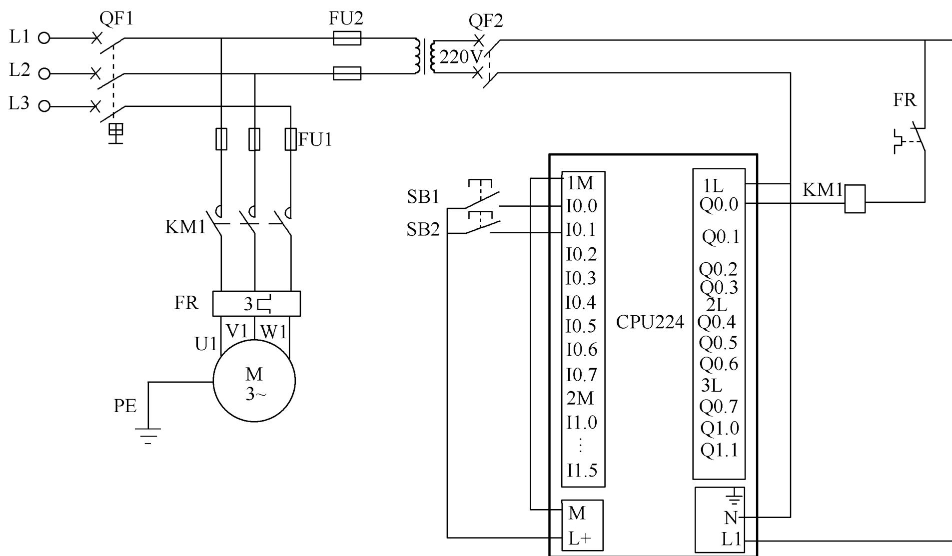


图 2. 3-2　PLC 系统接线原理图

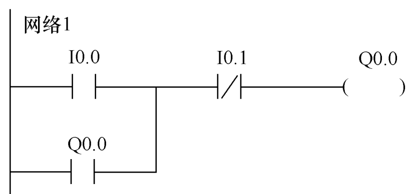


图 2. 3-3　参考程序 1

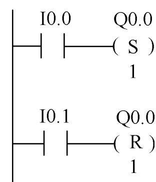


图 2. 3-4　参考程序 2

（4）程序输入与调试

1）程序输入。在编程软件中录入图 2. 3-3 或图 2. 3-4 所示的程序后再下载到 PLC 中。

2）程序调试。将 PLC 的输入端子 I0. 0 与电动机的起动按钮连接，I0. 1 与电动机的停止按钮连接；输出端子 Q0. 0 与控制电动机的接触器线圈连接。

① 电动机起动。接通电源，按下起动按钮后 I0. 0 闭合，Q0. 0 接通并置位保持，接触器 KM1 吸合，电动机运转。

② 电动机停止。按下停止按钮后 I0. 1 断开，Q0. 0 复位断电，接触器 KM1 断电释放，电动机停止运转。

实例 2　PLC 控制三相异步电动机正反转运行

1. 传统的正反转继电控制线路

（1）控制要求

在实际生产机械设备中，要满足生产机械运动部件的正、反两个方向运动，就得用电动机的正反转来控制。图 2. 3-5 所示为接触器联锁正反转控制线路，其控制要求如下。

1）按下正转起动按钮，电动机正向连续运转。

2）按下反转按钮，电动机反向连续运转。

3）按下停止按钮，电动机停止运转。

4）当电动机出现过载时，过载保护装置能立即使电动机停止运转。

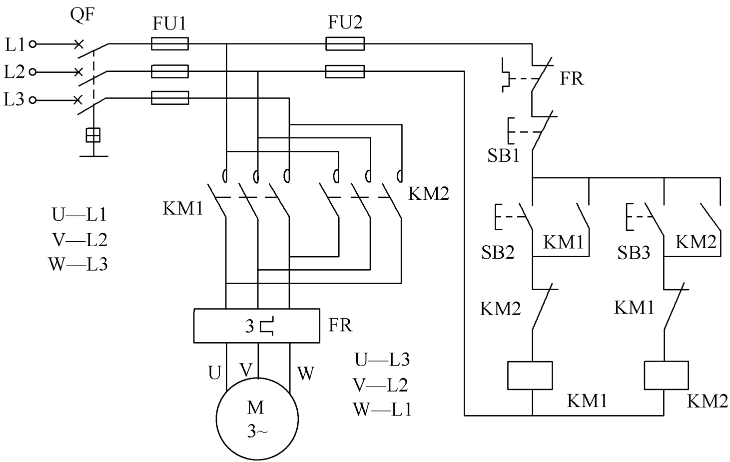
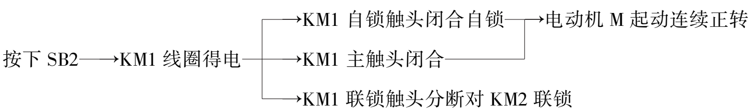


图 2. 3-5　接触器联锁正反转控制线路

（2）工作原理分析

接触器联锁正反转控制线路是利用两个交流接触器交替进行工作，通过改变电源接入电动机的相序，来实现电动机正反转控制。接触器联锁正反转控制线路工作原理分析如下。

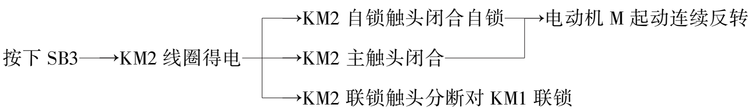
1）正转控制。



2）停止。

按下停止按钮 SB1→控制电路失电→KM1（或 KM2）主触头分断→电动机 M 失电停转

3）反转控制。



4）过载控制。当电动机发生过载时，热继电器动作，其常闭触点 FR 断开，切断控制回路电源，接触器线圈断电，其主触头分断，电动机停止运转。

2. PLC 控制电动机正反转线路

（1）I/O 地址通道分配

用 PLC 控制电动机正反转，其正反转控制线路的主电路不变，仅用 PLC 对控制电路进行改造。根据控制要求，确定 I/O 的个数，进行 I/O 的分配。本实例需要 4 个输入点，2 个输出点，见表 2. 3-2。

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 2. 3-6）

（3）程序设计

电动机正反转控制线路的梯形图程序如图 2. 3-7 所示。

表 2.3-2　PLC 的 I/O 配置

输入设备		输入继电器	输出设备		输出继电器
代号	功能		代号	功能	
SB1	正向起动按钮	I0.0	KM1	正向运行接触器	Q0.0
SB2	停止按钮	I0.1	KM2	反向运行接触器	Q0.1
SB3	反向起动按钮	I0.2			
FR	过载保护	I0.3			

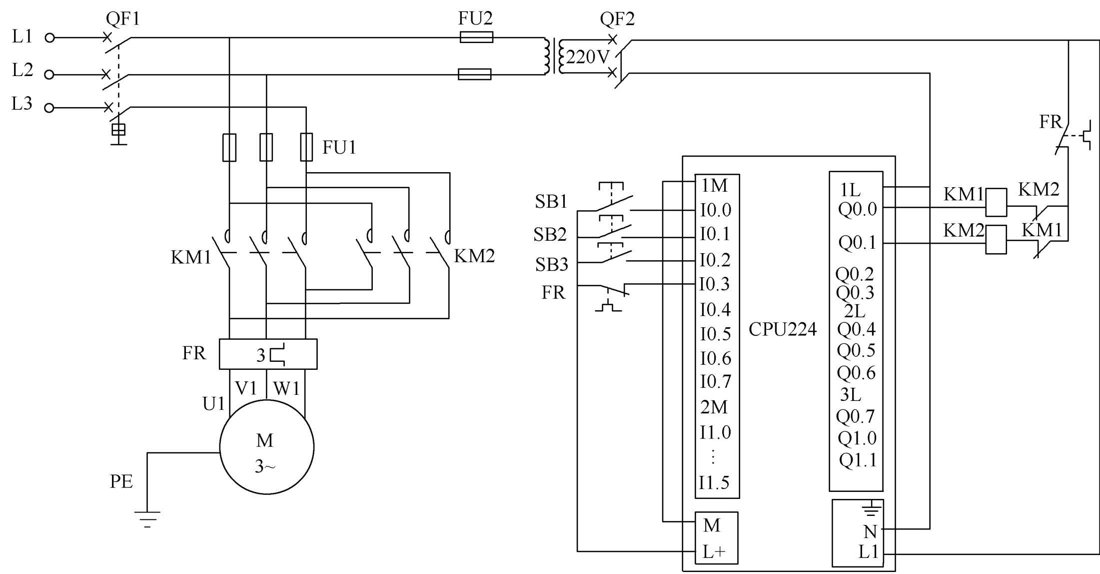


图 2.3-6　PLC 接线原理图

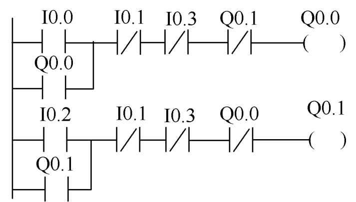


图 2.3-7　电动机正反转控制线路的梯形图程序

（4）程序输入与调试

1）程序输入。熟练地操作编程软件，能正确将编制的程序输入 PLC。

2）程序调试。按照被控设备的要求进行调试、修改，达到设计要求。

① 正转起动。按下正转起动按钮后 I0.0 闭合，Q0.0 接通并保持，接触器 KM1 吸合，电动机正转运行。

② 反转起动。按下反转起动按钮后 I0.2 闭合，Q0.1 接通并保持，接触器 KM2 吸合，电动机反转运行。

③ 电动机停止。按下停止按钮后 I0.1 断开，Q0.0（或 Q0.1）断电复位，接触器 KM1（或 KM2）断电释放，电动机停止运转。

④ 过载控制。当电动机过载时 I0.3 断开，Q0.0（或 Q0.1）断电复位，接触器 KM1（或 KM2）断电释放，电动机停止运转。

实例 3　PLC 控制三相异步电动机Y-△减压起动

1. 传统的继电器控制Y-△减压起动线路

（1）控制要求

电源容量在 180kVA 以上，电动机容量在 7kW 以下的三相异步电动机可采用直接起动。一般大于 7.5kW 的交流异步电动机，在起动时常采用Y-△减压起动。

1）按下起动按钮后，电动机先作星形联结起动，经延时一段时间后，自动切换成三角形联结运转。

2）按下停止按钮后，电动机停止运转。

（2）工作原理分析

图 2.3-8 所示为时间继电器自动控制Y-△减压起动线路，该线路由三个接触器、一个热继电器、一个时间继电器和两个按钮组成。其中，接触器 KM 作引入电源用；接触器 KMY和 KM△分别作Y减压起动用和△运行用；时间继电器 KT 用作控制Y减压起动时间和完成Y-△自动切换；SB1 是起动按钮；SB2 是停止按钮；FU1 做主电路的短路保护；FU2 做控制电路的短路保护；FR 做过载保护。

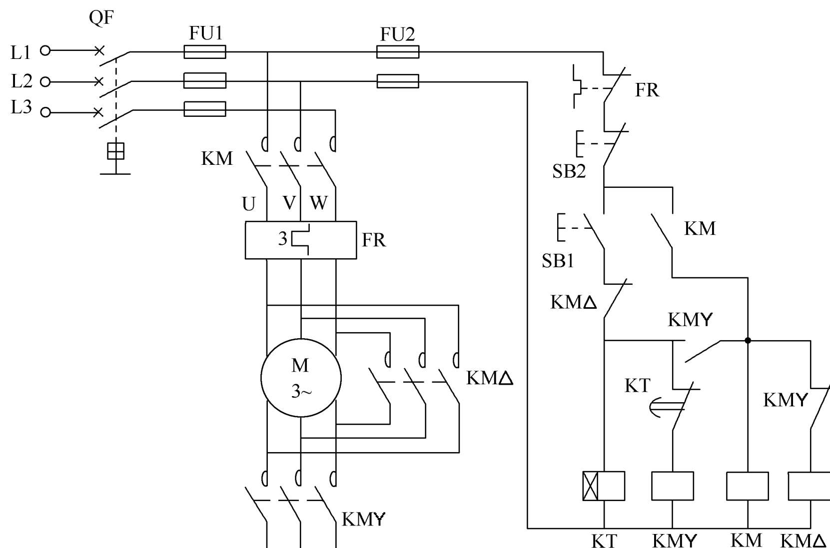


图 2.3-8　时间继电器自动控制Y-△减压起动线路

Y-△减压起动控制线路工作原理如下。

合上电源开关 QF。

减压起动：



停止时，按下 SB2 即可实现。

2. PLC 控制三相异步电动机Y-△减压起动

（1）I/O 地址通道分配

根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 3 个输入点，3 个输出点，见表 2.3-3。

表 2.3-3　PLC 的 I/O 配置

输入设备		输入继电器	输出设备		输出继电器
代号	功能		代号	功能	
SB1	起动按钮	I0.0	KM1	主接触器	Q0.0
SB2	停止按钮	I0.1	KM2	Y接触器	Q0.1
FR	过载保护	I0.2	KM3	△接触器	Q0.2

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 2.3-9）

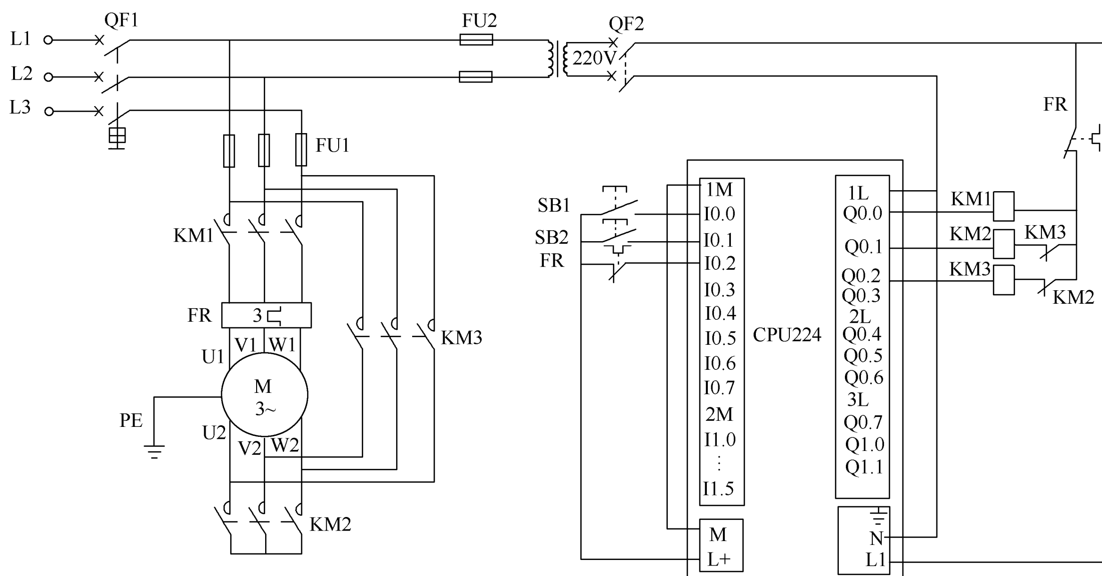


图 2.3-9　PLC 系统接线原理图

（3）程序设计

Y-△减压起动控制线路的参考梯形图如图 2.3-10 所示。

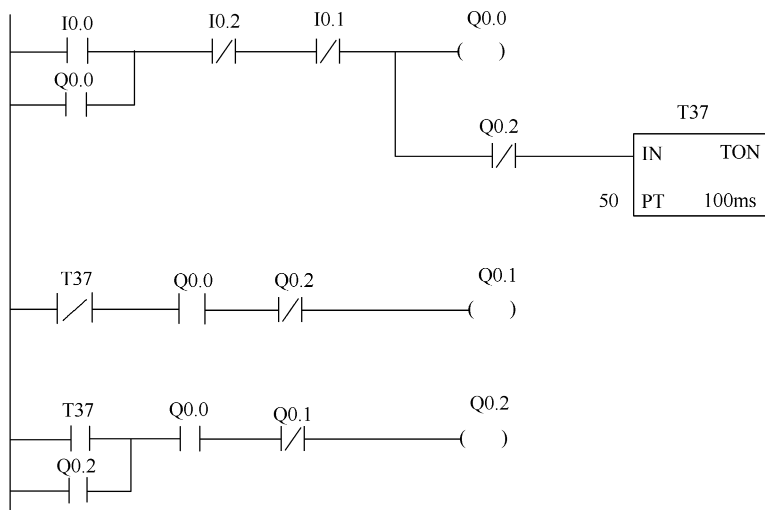


图 2.3-10　Y-△减压起动控制梯形图

（4）程序输入与调试

1）程序输入。熟练地操作编程软件，能正确将编制的程序输入 PLC。

2）程序调试。按照被控设备的要求进行调试、修改，达到设计要求。

① 减压起动。按下起动按钮后 I0.0 闭合，Q0.0 接通并保持，接触器 KM1 吸合，Q0.1 接通并保持，接触器 KM2 吸合，电动机连接成星形起动运行；同时，T37 定时器开始延时。

② T37 延时时间到 5s 后，Q0.1 断电，接触器 KM2 释放切断星形回路；同时 Q0.2 接通并保持，接触器 KM3 接通，电动机连接成三角形运转。

③ 电动机停止。按下停止按钮后 I0.1 断开，Q0.0、Q0.2 断电复位，接触器 KM1 和 KM3 断电释放，电动机停止运转。

④ 过载控制。当电动机过载时 I0.2 断开，Q0.0 断电复位，接触器 KM1 和 KM2 断电释放，电动机停止运转。

实例 4　PLC 控制三台电动机顺序起停

1. 控制要求

有三台电动机 M1、M2、M3 顺序控制，当按下起动按钮 SB1 时，PLC 的输入信号 I0.0 有效，电动机 M1 起动；M1 起动后，延时 5s 第二台电动机 M2 起动；M2 起动后，再延时 5s 第三台电动机 M3 起动。按下停止按钮 SB2 时，PLC 的输入信号 I0.1 有效，三台电动机同时停止。

2. 用 PLC 控制三台电动机顺序起停

（1）I/O 地址通道分配

根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 3 个输入点，3 个输出点，见表 2.3-4。

表 2.3-4　PLC 的 I/O 配置

输入设备		输入继电器	输出设备		输出继电器
代号	功能		代号	功能	
SB1	起动按钮	I0.0	KM1	接触器 1	Q0.0
SB2	停止按钮	I0.1	KM2	接触器 2	Q0.1
FR	过载保护	I0.2	KM3	接触器 3	Q0.2

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 2.3-11）

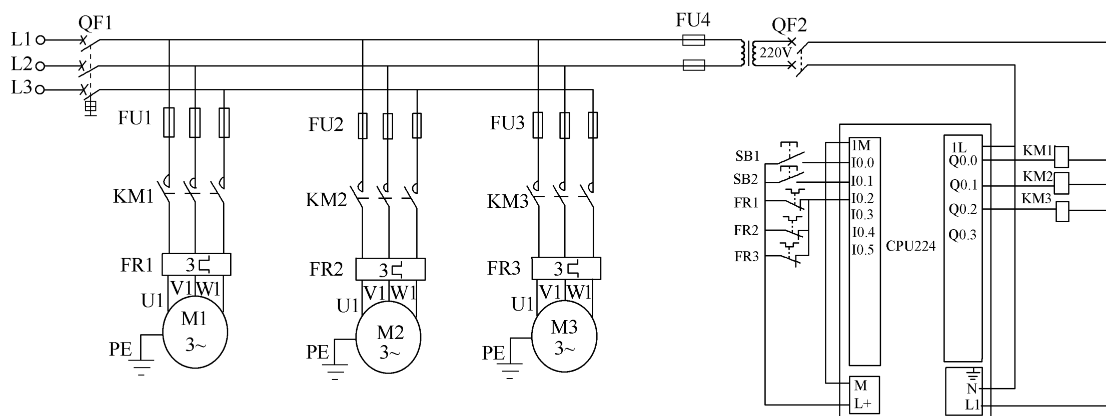


图 2.3-11　PLC 系统接线原理图

（3）程序设计

PLC 控制三台电动机顺序起停的参考梯形图如图 2.3-12 所示。

（4）程序输入与调试

1）程序输入。熟练地操作编程软件，能正确将编制的程序输入 PLC。

2）程序调试。按照被控设备的要求进行调试、修改，达到设计要求。

① 当 I0.0 接通后，Q0.0 置位，Q0.0 的常开触点接通 T37 定时器，延时 5s 后 Q0.1 置位，Q0.1 的常开触点接通 T38 定时器，延时 5s 后 Q0.2 置位，三台电动机延时顺序起动。

② 当 I0.1 接通后，Q0.0、Q0.1 和 Q0.2 同时复位。

实例 5　电动机位置自动往返控制

1. 控制要求

在实际工作中，经常要求电动机进行自动延时循环控制。按下正转起动按钮，电动机正向起动向右运转，运行到终端碰到限位开关 SQ2，电动机停止并延时 5s 向左返回，碰到限位开关 SQ1，电动机停止并延时 5s 再一次向右运行，如此自动往返；按下反转按钮，电动机反向起动运转，也按照上述要求自动往返。按下停止按钮后，电动机停止运转。电动机具有过载保护，循环示意图如图 2.3-13 所示。

2. PLC 控制电动机自动往返

（1）I/O 地址通道分配

根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 6 个输入点，2 个输出点，见表 2.3-5。

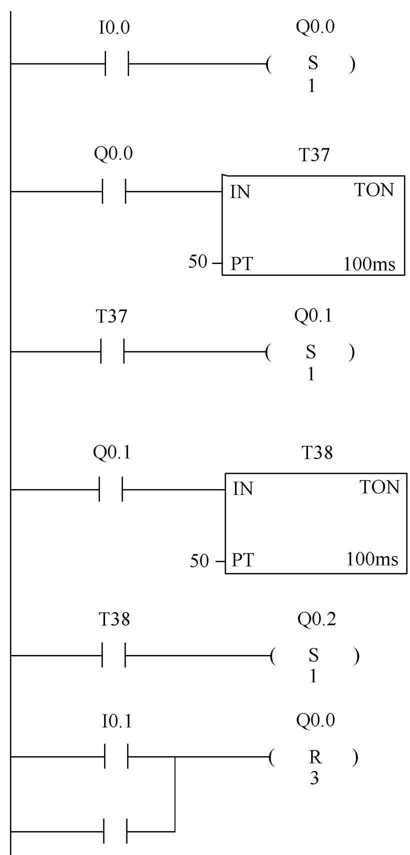


图 2. 3-12　三台电动机顺序起停梯形图

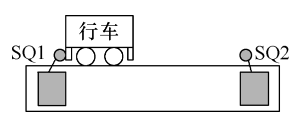


图 2. 3-13　自动延时循环控制示意图

表 2. 3-5　PLC 的 I/O 配置

输 入 设 备		输入继电器	输 出 设 备		输出继电器
代号	功能		代号	功能	
SB1	正向起动按钮	I0. 0	KM1	正向运行接触器	Q0. 0
SB2	停止按钮	I0. 1	KM2	反向运行接触器	Q0. 1
SB3	反向起动按钮	I0. 2			
FR	过载保护	I0. 3			
SQ1	反向限位开关	I0. 4			
SQ2	正向限位开关	I0. 5			

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 2. 3-14）

（3）程序设计

电动机自动往返循环控制线路的参考梯形图如图 2. 3-15 所示。

（4）程序输入与调试

1）程序输入。熟练地操作编程软件，能正确将编制的程序输入 PLC。

2）程序调试。按照被控设备的要求进行调试、修改，达到设计要求。

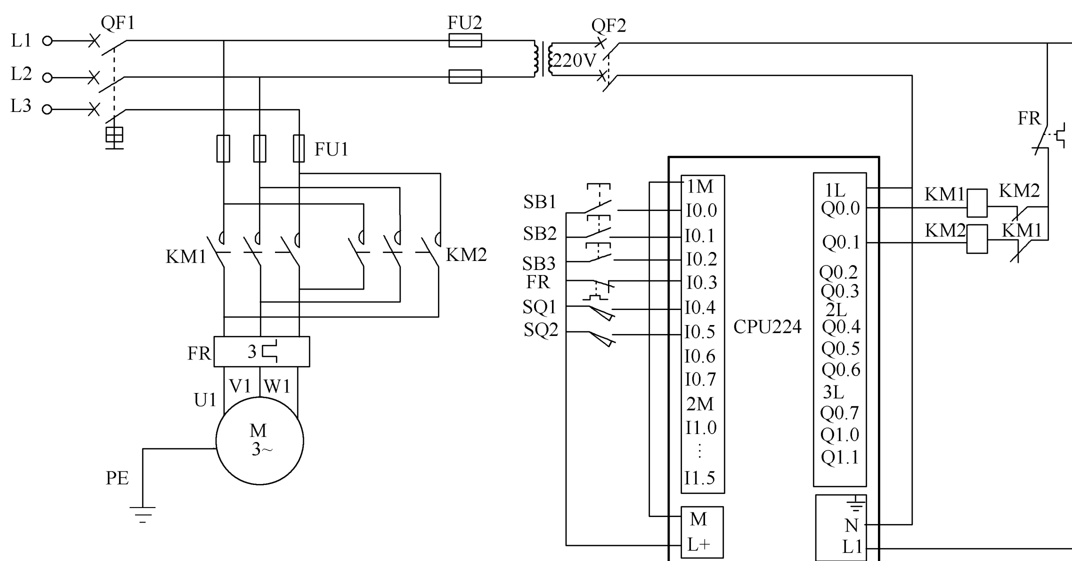


图 2. 3-14　PLC 系统接线原理图

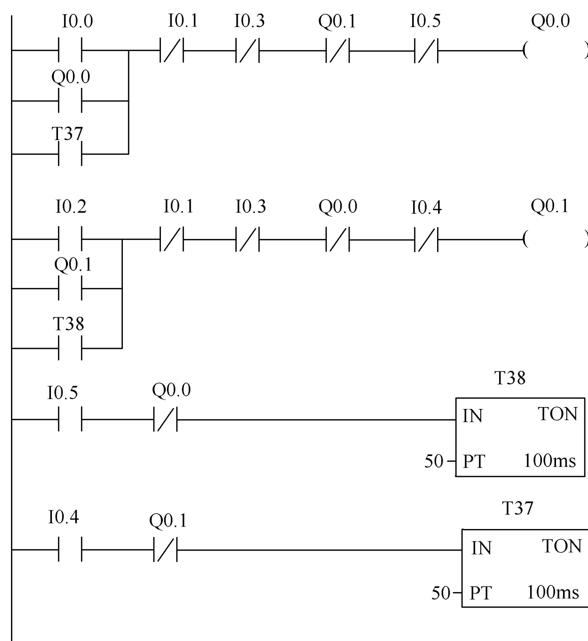


图 2. 3-15　电动机自动往返循环控制线路的梯形图

① 正向运行。按下正向起动按钮后 I0. 0 闭合，Q0. 0 接通并保持，接触器 KM1 吸合，电动机正向起动运行。当运行到碰到正向限位开关 SQ2 时，切断正向运行回路，电动机停止运行，同时 T38 定时器开始延时，为反向接通做准备。

② 反向运行。T38 定时器延时 5s 后接通反向运行（或接通反向运行按钮 I0. 2），Q0. 1 接通，接触器 KM2 吸合，电动机反向运行。当运行到碰到反向限位开关 SQ1 时，切断反向运行回路，电动机停止运行，同时 T37 定时器开始延时。

③ T37 延时 5s 后接通正向运行，如此循环。

④ 停止运行。按下停止按钮 SB2 时，电动机停止运行。

实例 6　长定时的 PLC 控制

1. 控制要求

用 PLC 控制长定时电铃。

2. 用 PLC 控制长定时电铃

（1）I/O 地址通道分配

根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 1 个输入点，1 个输出点，见表 2.3-6。

表 2.3-6　PLC 的 I/O 配置

输入设备		输入继电器	输出设备		输出继电器
代号	功能		代号	功能	
SA	起动开关	I0.0		电铃	Q0.0

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 2.3-16）

（3）程序设计

在 S7-200 PLC 中定时器最长定时时间不到 1h，但在实际应用中，经常需要几小时的定时控制，这样一个定时器是不能完成任务的，下面介绍两种长延时控制程序。

参考程序 1：多个定时器组合实现长延时控制。

图 2.3-17 所示为利用两个定时器组成的长延时控制程序。图中是两级定时器串联使用，T37 延时 $T_1=2400s$，T38 延时 $T_2=3000s$，总计延时 $T=T_1+T_2=5400s$。

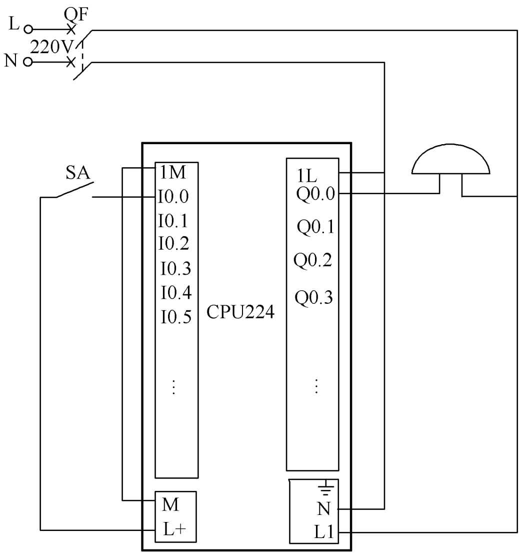


图 2.3-16　PLC 系统接线原理图

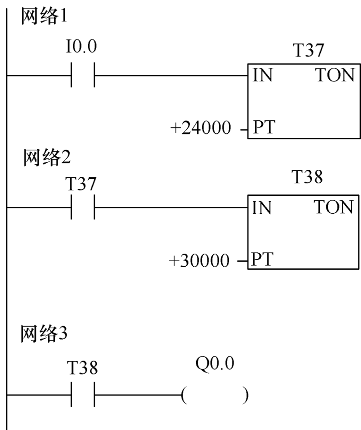


图 2.3-17　定时器串联长延时控制程序

参考程序 2：定时器与计数器组合实现长延时控制。

图 2. 3-18 所示为利用定时器与计数器组合应用来实现长延时电路的控制程序，图中实现 2000 × 10s = 20 000s 的延时控制。

4. 程序输入与调试

能正确将编制的两个程序分别输入 PLC，并按照要求进行调试、修改，达到设计要求。

实例 7　闪光灯的 PLC 控制

1. 控制要求

接通开关 SA 时 I0. 0 闭合，定时器产生一个 1s 通、2s 断的闪烁信号，来控制 Q0. 0 通 1s、断 2s，实现报警灯的闪烁控制。

2. 用 PLC 控制闪光灯

（1） I/O 地址通道分配

根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 1 个输入点，1 个输出点，见表 2. 3-7。

表 2. 3-7　PLC 的 I/O 配置

输入设备		输入继电器	输出设备		输出继电器
代号	功能		代号	功能	
SA	起动开关	I0. 0	EL	闪光灯	Q0. 0

（2） 根据控制要求，设计并绘制 PLC 系统接线原理图（见图 2. 3-19）

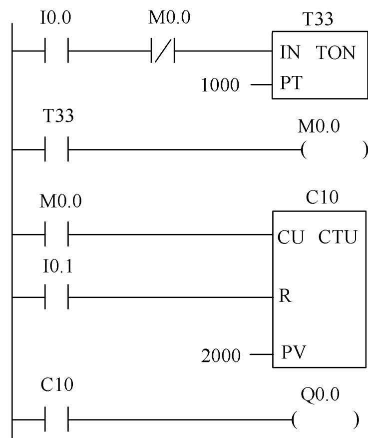


图 2-3-18　定时器与计数器组合实现长延时控制程序

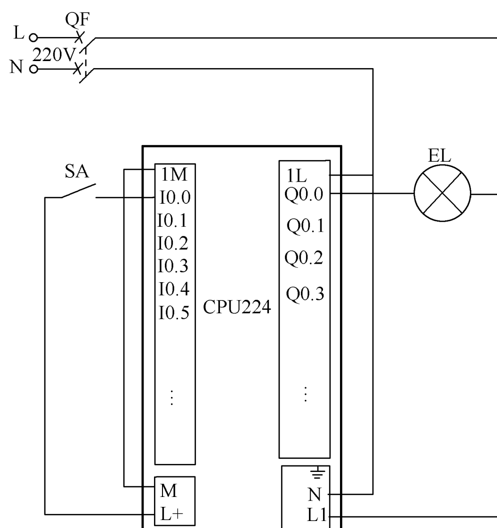


图 2. 3-19　PLC 系统接线原理图

（3） 程序设计

闪烁控制程序实际是一个振荡控制程序，图 2. 3-20 所示为一种典型的闪烁控制程序。

（4） 程序输入与调试

能正确将编制的程序输入 PLC，并按照要求进行调试、修改，达到设计要求。

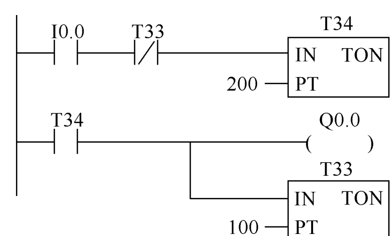


图 2. 3-20　闪烁控制程序

实例 8　单按钮 PLC 控制电动机起停

1. 控制要求

利用一只按钮完成电动机的起停控制，第一次按下时，电动机起动；第二次按下该按钮时，电动机停止；第三次按下该按钮时，电动机又起动，如此循环。

2. 单按钮 PLC 控制电动机起停

（1）I/O 地址通道分配

根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 1 个输入点，1 个输出点，见表 2. 3-8。

表 2. 3-8　PLC 的 I/O 配置

输 入 设 备		输入继电器	输 出 设 备		输出继电器
代号	功能		代号	功能	
SB	按钮	I0. 0	KM	接触器	Q0. 0

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 2. 3-21）

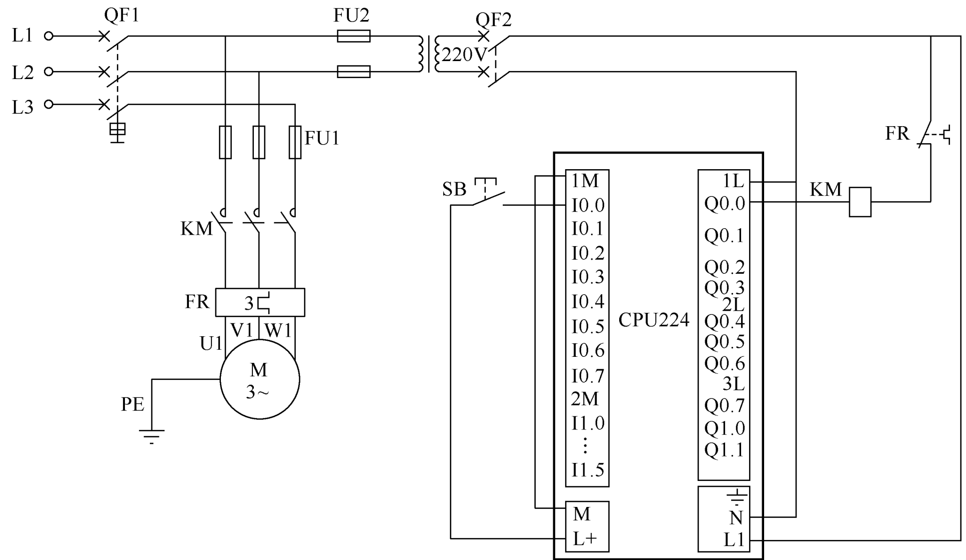


图 2. 3-21　PLC 系统接线原理图

（3）程序设计

图 2. 3-22 为单按钮 PLC 控制电动机的起停梯形图。

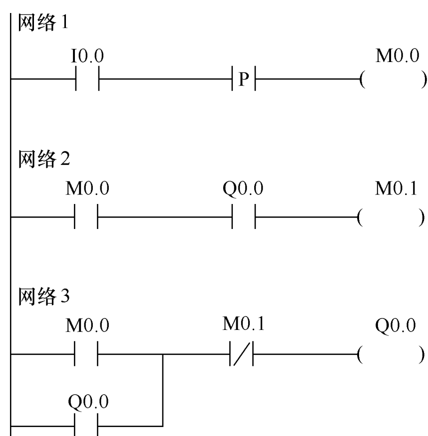


图 2. 3-22　单按钮 PLC 控制电动机的起停梯形图

（4）程序输入与调试

能正确将编制的程序输入 PLC，并按照要求进行调试、修改，达到设计要求。

① 当第一次按下按钮时 I0. 0 闭合，其上升沿脉冲使继电器 M0. 0 闭合一个扫描周期，Q0. 0 输出线圈接通并保持，KM 接触器吸合，电动机起动运转。

② 当第二次按下该按钮时 I0. 0 闭合，M0. 0 再次接通一个扫描周期并与已闭合的 Q0. 0 辅助触点共同触发继电器 M0. 1，使其接通一个扫描周期，辅助常闭触点 M0. 1 断开使 Q0. 0 输出线圈断电复位，电动机停止运行。

③ 第三次按下该按钮，电动机再次起动，如此循环。

第三章

步进顺序控制

第一节　顺序控制及顺序功能图

一、顺序控制概述

梯形图的设计方法一般称为经验设计法，经验设计法没有一套固定的步骤可循，具有很大的试探性和随意性。在设计复杂系统的梯形图时，用大量的中间单元来完成记忆、联锁和互锁等功能，由于需要考虑的因素很多，这些因素又往往交织在一起，分析起来非常困难。

顺序控制设计法是一种先进的设计方法，很容易被初学者接受，有经验的工程师使用顺序控制设计法，也会提高设计的效率，程序调试、修改和阅读也更方便。

所谓顺序控制，就是按照生产工艺预先规定的顺序，在各个输入信号的作用下，根据内部状态和时间的顺序，在生产过程中各个执行机构自动地有序地进行操作。

例如，某设备有三台电动机，控制要求是按下起动按钮，第一台电动机 M1 起动；M1 运行 5s 后，第二台电动机 M2 起动；M2 运行 15s 后，第三台电动机 M3 起动；按下停止按钮，三台电动机全部停机。

现将三台电动机顺序控制的各个控制步骤用工序表示，并依工作顺序将工序连接起来，如图 3. 1-1 所示。该工序图具有以下特点：

1）复杂的控制任务或工作过程分解成若干个工序。

2）各工序的任务明确而具体。

3）各工序间的联系清楚，可读性很强，能清晰地反映整个控制过程，并给编程人员清晰的编程思路。

二、顺序功能图

顺序功能图是用来描述顺序控制过程的，任何一个顺序控制过程都可以分解为若干个阶段，称为步或状态，每个状态都有不同的动作，当相邻两状态之间的转换条件得到满足时，就将实现转换，即由上一个状态转换到下一个状态。根据图 3. 1-1 可以画出三台电动机顺序控制的功能图，如图 3. 1-2 所示。

1. 顺序功能图的组成

顺序功能图（简称功能图）又叫状态流程图或状态转移图。它由步、有向连线、转换、转换条件和动作（或命令）组成。

（1）步　步是控制系统中的一个相对稳定的状态，它是根据输出量的状态变化来划分

的，在任何一步内，各个输出量的ON/OFF状态不变，但是相邻步的输出量总的状态是不同的。在顺序功能图中分为中间步和初始步，如图3.1-3所示。

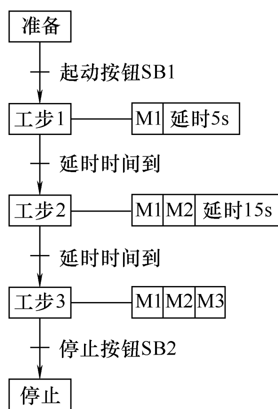


图3.1-1　工序图

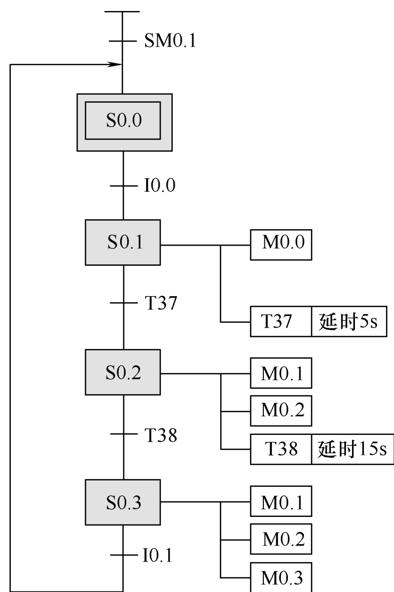


图3.1-2　顺序功能图

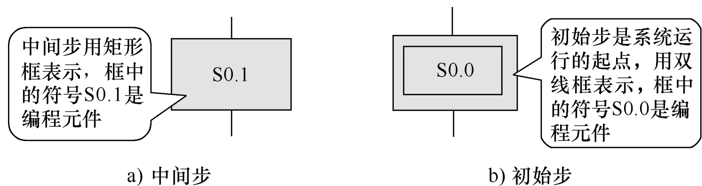


图3.1-3　中间步和初始步

（2）有向连线　步与步之间的有向连线用来表示步的活动状态和进展方向。从上到下和从左到右这两个方向上的箭头可以省略，其他方向上必须加上箭头用来注明步的进展方向。图3.1-4中流程方向始终向下，因而省略了箭头。

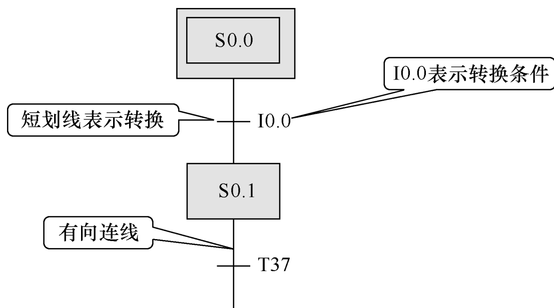


图3.1-4　有向连线

（3）转换　转换用来将相邻两步分隔开，用与有向连线垂直的短划线表示，如图3.1-4所示。

（4）转换条件　转换条件是与转换有关的逻辑命题，转换条件可以用文字语言、布尔代数表达式或图形符号标注在表示转换的短线旁边，如图3.1-4所示。

（5）动作（或命令）　一个步表示控制过程中的稳定状态，它可以对应一个或多个动作。可以在步右边加一个矩形框，在框中用简明的文字说明该步对应的动作。图3.1-5a表

示一个步对应一个动作，图 3.1-5b 表示一个步对应多个动作。

2. 设计顺序功能图的方法和步骤

下面以小车往返的控制来说明顺序功能图的绘制方法。

图 3.1-6 为小车自动往返的示意图，其控制要求如下：

当按下起动按钮 SB 后，电动机 M 正转（由输出线圈 Q0.0 驱动），小车向右前进，碰到限位开关 SQ1 后，电动机 M 反转（由输出线圈 Q0.1 驱动），小车向左后退，当小车后退碰到限位开关 SQ2 后，小车电动机 M 停转，小车停车 3s 后，第二次向右前进，碰到限位开关 SQ3，再次后退，当后退再次碰到限位开关 SQ2 时，小车停止。

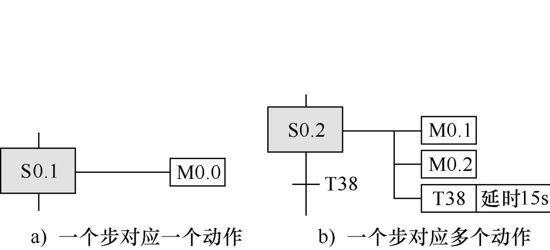


图 3.1-5　动作（或命令）

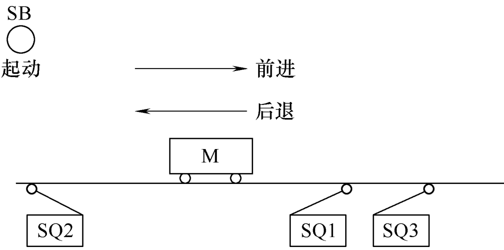


图 3.1-6　小车往返控制示意图

1）将整个控制过程按任务要求分解成若干个工序，其中的每一个工序都对应一个状态（步），并分配辅助状态继电器，见表 3.1-1。

表 3.1-1　状态继电器分配表

任　　务	状态继电器	任　　务	状态继电器
初始状态	S0.0	延时 3s	S0.3
前进	S0.1	再前进	S0.4
后退	S0.2	再后退	S0.5

2）搞清楚每个状态的功能与作用，见表 3.1-2。状态的功能是通过 PLC 驱动各种负载来完成的，负载可由状态元件直接驱动，也可由其他软触点的逻辑组合驱动。

表 3.1-2　状态继电器的功能与作用表

状态继电器	功能与作用	状态继电器	功能与作用
S0.0	PLC 上电做好工作准备	S0.3	延时 3s（定时器 T37，设定 3s）
S0.1	前进（输出 Q0.0，驱动电动机 M 正转）	S0.4	前进（输出 Q0.0，驱动电动机 M 正转）
S0.2	后退（输出 Q0.1，驱动电动机 M 反转）	S0.5	后退（输出 Q0.1，驱动电动机 M 反转）

3）找出每个状态的转换条件和方向，即在什么条件下将下一个状态“激活”，见表 3.1-3。状态的转换条件可以是单一的触点，也可以是多个触点的串、并联电路的组合。

表 3.1-3　状态的转换条件表

状态继电器	转 换 条 件	状态继电器	转 换 条 件
S0.0	SM0.1	S0.3	SQ2
S0.1	SB	S0.4	T37
S0.2	SQ1	S0.5	SQ3

4）根据控制要求或工艺要求，画出顺序功能图，如图 3.1-7 所示。

3. 使用注意事项

在使用顺序功能图进行编程时，要注意以下几个方面。

1）两个相邻步与步之间不能直接相连，必须用转换分开。

2）两个转换与转换之间不能直接相连，必须用步分开。

3）步与转换、转换与步之间的连线采用有向连线，画功能图的顺序一般是从上向下或从左到右，正常顺序时可以省略箭头，否则必须加箭头。

4）一个功能图至少应有一个初始步，如没有它系统将无法开始和返回。

5）必须用初始化脉冲 SM0.1 常开触点作为转换条件，将初始步转化为活动步。

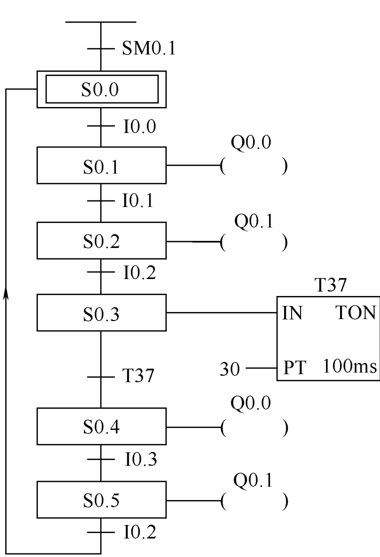


图 3-1-7　顺序功能图

三、步进顺控指令

在顺序控制或步进控制中，常常将控制过程分为若干个顺序控制继电器（SCR）段，一个 SCR 段有时也称为一个控制功能步，简称步。每个 SCR 段都是一个相对稳定的状态，都有段开始、段结束、段转换，在 S7－200 中有三条简单的 SCR 指令与之对应，见表 3.1-4。

表 3.1-4　顺控（SCR）指令

梯　形　图	语　句　表	描　　述
SCR	LSCR S_ bit	SCR 程序段开始
SCRT	SCRT S_ bit	SCR 转换
SCRE	SCRE	SCR 程序段结束

1. 段开始指令 LSCR

段开始指令的功能是标记一个 SCR 段（或一个步）的开始，其操作数是状态继电器 SI. Q（如 S0.0），SI. Q 是当前 SCR 段的标志位，当 SI. Q 为 1 时，允许 SCR 段工作。

2. 段转换指令 SCRT

段转换指令的功能是将当前的 SCR 段切换到下一个 SCR 段，其操作数是下一个 SCR 段的标志位 SI. Q（如 S0.1）。当允许输入有效时，进行切换，即停止当前 SCR 段工作，起动下一个 SCR 段工作。

3. 段结束指令 SCRE

段结束指令的功能是标记一个 SCR 段（或一个步）的结束。每个 SCR 段必须使用段结束指令来表示该 SCR 段的结束。

4. 梯形图表示法

在梯形图中，段开始指令以功能框的形式编程，指令名称为 SCR，如图 3.1-8 所示；段

转换和段结束指令以线圈形式编程，如图 3. 1-9 所示。

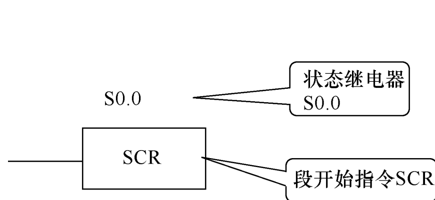


图 3. 1-8　段开始指令表示法

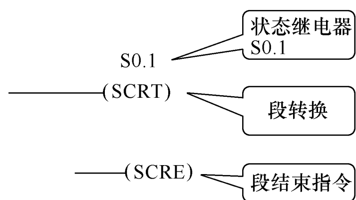


图 3. 1-9　段转换和段结束指令表示法

5. 使用注意事项

1）SCR 指令的操作数（或编程元件）只能是状态继电器 SI. Q。

2）一个状态继电器 SI. Q 作为 SCR 段标志位，可以用于主程序、子程序或中断程序中，但是只能使用一次，不能重复使用。

3）在一个 SCR 段中，禁止使用循环指令 FOR/NEIT、跳转指令 JMP/LBL 和条件结束指令 END。

第二节　单流程结构步进顺序控制

顺序功能图有三种基本结构：单流程、选择流程和并行流程，另外还有复合、循环、跳转和重复等流程结构形式，不同的结构流程，其特点和应用方法也有区别，下面重点介绍前三种流程。

一、单流程结构顺序功能图

单流程结构的顺序功能图，只有一个转换条件并转向一个分支，是最基本的结构流程。它由顺序排列、依次有效的状态序列组成，每个状态的后面只跟一个转换条件，每个转换条件后面也只连接一个状态，如图 3. 2-1 所示。

在图 3. 2-1 中，当 PLC 上电时，SM0. 1 产生一个扫描周期的脉冲使 S0. 0 置为活动步。当起动条件满足时，状态 S0. 1 有效。若转换条件 I0. 1 接通，状态将从 S0. 1 转换到 S0. 2，当转换完成后，S0. 1 同时复位。同样，当状态 S0. 2 有效时，若转换条件 T37 接通，将从 S0. 2 转换到 S0. 3，转换完成，S0. 2 同时复位。依次类推，直至最后一个状态。

二、单流程结构的编程

利用顺序功能图对控制任务进行编程，通常有以下六个步骤。

1）任务分解；

2）I/O 分配；

3）动作设置；

4）转换条件设置；

5）绘制顺序功能图；

6）顺序功能图的转换。

在进行顺序控制编程时，必须遵循上述已经指出的方法与步骤。这里重点介绍顺序功能

图与梯形图的转换。例如，将图 3. 2-1 所示的功能图转换成如图 3. 2-2 所示的梯形图。

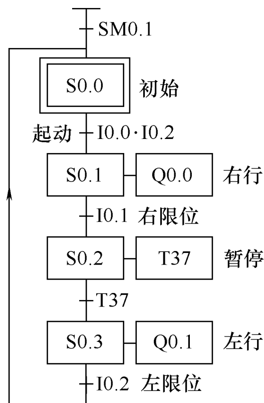


图 3. 2-1　单流程的顺序功能图

梯形图	说明
SM0.1 ─┤ ├─ (S) S0.0, 1	首次扫描，置位初始步S0.0
S0.0 SCR	初始步的SCR段开始
I0.0 ─┤ ├─ I0.2 ─┤/├─ (SCRT) S0.1	按起动按钮时转换到右行步S0.1
(SCRE)	初始步的SCR段结束
S0.1 SCR	右行步的SCR段开始
SM0.0 ─┤ ├─ ( ) Q0.0	小车右行
I0.1 ─┤ ├─ (SCRT) S0.2	碰到右限位开关，转换到暂停步S0.2
(SCRE)	右行步的SCR段结束
S0.2 SCR	暂停步的SCR段开始
SM0.0 ─┤ ├─ T37: IN TON, 30 ─ PT 100ms	延时3s
T37 ─┤ ├─ (SCRT) S0.3	延时时间到，转换到左行步S0.3
(SCRE)	暂停步的SCR段结束
S0.3 SCR	左行步的SCR段开始
SM0.0 ─┤ ├─ ( ) Q0.1	小车左行
I0.2 ─┤ ├─ (SCRT) S0.0	碰到左限位开关，返回到初始步S0.0
(SCRE)	左行步的SCR段结束

图 3. 2-2　梯形图

第三节　选择结构步进顺序控制

一、选择结构顺序功能图

选择结构的顺序功能图，要按不同转换条件选择转向不同分支，执行不同分支后再根据不同转换条件汇合到同一分支，如图 3. 3-1 所示。

二、选择结构的编程

选择结构的编程与一般编程一样，也必须遵循上述已经指出的方法。无论是从分支状态向各个流程分支散转时，还是从各个分支状态向汇合状态汇合时，都要正确使用这些方法。

1. 选择性分支的编程

在图 3. 3-1 中，S0. 0 称为分支状态，它下面有两个分支，根据不同的转换条件 I0. 0 和

I0.2 来选择转向其中的一个分支，这两个分支不能同时被选中。当 I0.0 接通时，状态将转换到 S0.1，而当 I0.2 接通时，状态将转换到 S0.2，所以要求转换条件 I0.0 和 I0.2 不能同时闭合。

2. 选择汇合的编程

在图 3.3-1 中，S0.3 称为汇合状态，状态 S0.1 或 S0.2 根据各自的转换条件 I0.1 或 I0.3 向汇合状态转换。一旦状态 S0.3 接通时，前一状态 S0.1 或 S0.2 就自动复位。

3. 顺序功能图与梯形图的转换

将选择结构的顺序功能图转换为梯形图时，关键是对分支和汇合状态的处理，如图 3.3-2 所示。

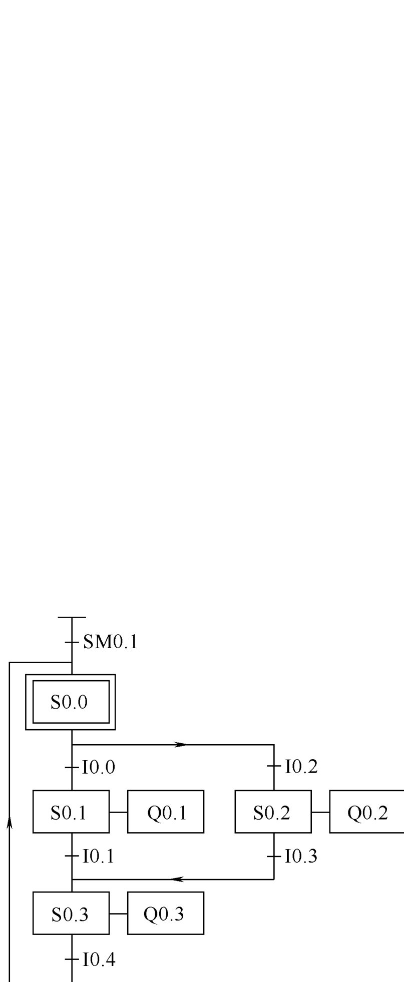


图 3.3-1　选择结构的顺序功能图

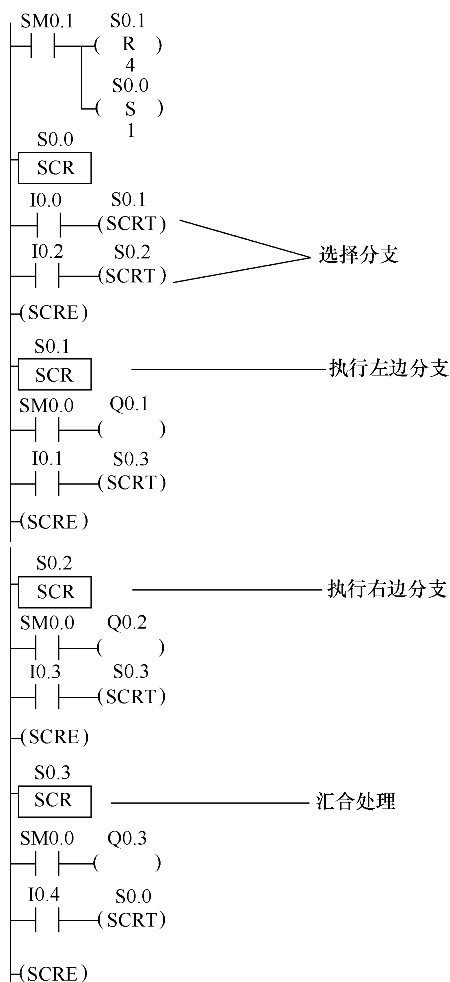


图 3.3-2　梯形图

（1）分支状态的处理　先进行分支状态 S0.1 的驱动连接，依次按转换条件置位各分支的首转换状态，再从左至右对首转换状态先负载驱动，后转换处理。

（2）汇合状态的处理　先进行汇合前各分支的最后一个状态和汇合状态 S0.3 的驱动连接，再从左至右对汇合状态进行转换连接。

第四节　并行结构步进顺序控制

一、并行结构顺序功能图

并行结构的顺序功能图，按同一转换条件同时转向几个分支并激活，执行不同的分支后再汇合到同一分支形成一个汇合流。用水平双线来表示并行分支，上面一条表示并行分支的开始，下面一条表示并行分支的结束，如图 3.4-1 所示。

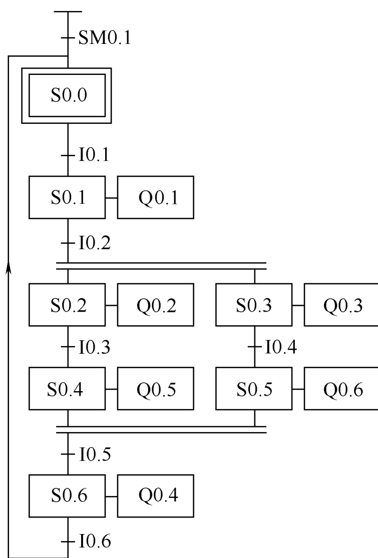


图 3.4-1　并行结构的顺序功能图

二、并行结构的编程

并行结构状态的编程与一般状态编程一样，先进行负载驱动，后进行转换处理，转换处理从左到右依次进行。无论是从分支状态向各个流程分支并行转换时，还是从各个分支状态向汇合状态同时汇合时，都要正确使用这些规则。

1）在图 3.4-1 中，S0.1 称为分支状态，它下面有两个分支，当转换条件 I0.2 接通时，两个分支将同时被选中，并同时并行运行。

2）在图 3.4-1 中，S0.6 称为汇合状态，当两个分支都执行到各自最后状态时，S0.4 和 S0.5 都接通，此时，S0.4 和 S0.5 处于等待状态不能自行复位，需用复位指令来完成，若转换条件 I0.5 接通，将一起转入汇合状态 S0.6。

3）顺序功能图与梯形图的转换。将并行结构的顺序功能图转换为梯形图时，关键是对并行分支和并行汇合的编程处理，如图 3.4-2 所示。

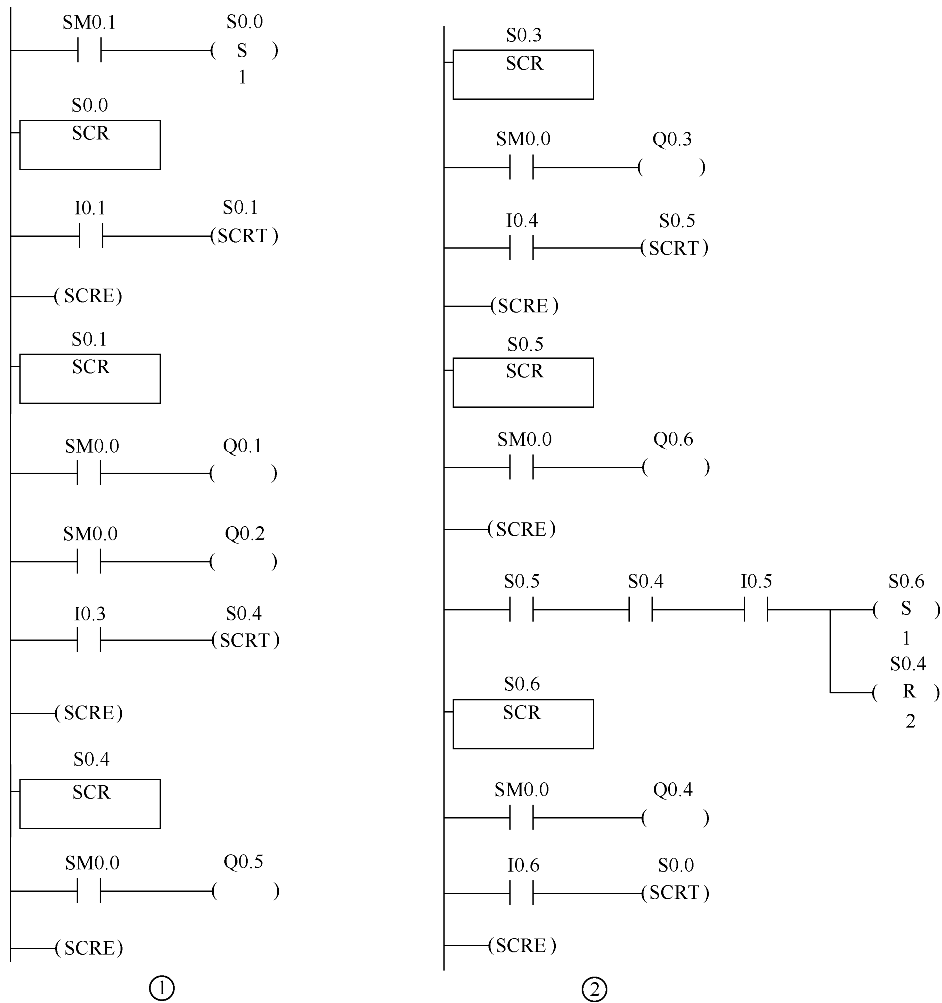


图 3. 4-2　梯形图

第五节　步进顺序控制的综合应用实例

实例 1　简易红绿灯控制系统

1. 控制要求

当按下起动按钮后，B 车道通行，绿灯亮，5s 后转为黄灯亮，黄灯亮 2s 后 B 车道禁行，A 车道通行，A 车道绿灯亮 5s 转为黄灯亮，2s 后 A 车道禁行，B 车道通行，依次循环。

2. 用 PLC 控制简易红绿灯的操作步骤

（1）根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配（见表 3. 5-1）

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 3. 5-1）

（3）程序设计

1）设计顺序功能图。根据红绿灯的状态转换规律和转换条件，设计顺序功能图，如图 3. 5-2所示。

表 3. 5-1　红绿灯控制系统的 I/O 分配表

输　入　点	输入设备及作用	输　出　点	输出设备及作用
I0. 0	起动开关 S	Q0. 0	A 向绿灯
		Q0. 1	A 向黄灯
		Q0. 2	A 向红灯
		Q0. 3	B 向绿灯
		Q0. 4	B 向黄灯
		Q0. 5	B 向红灯

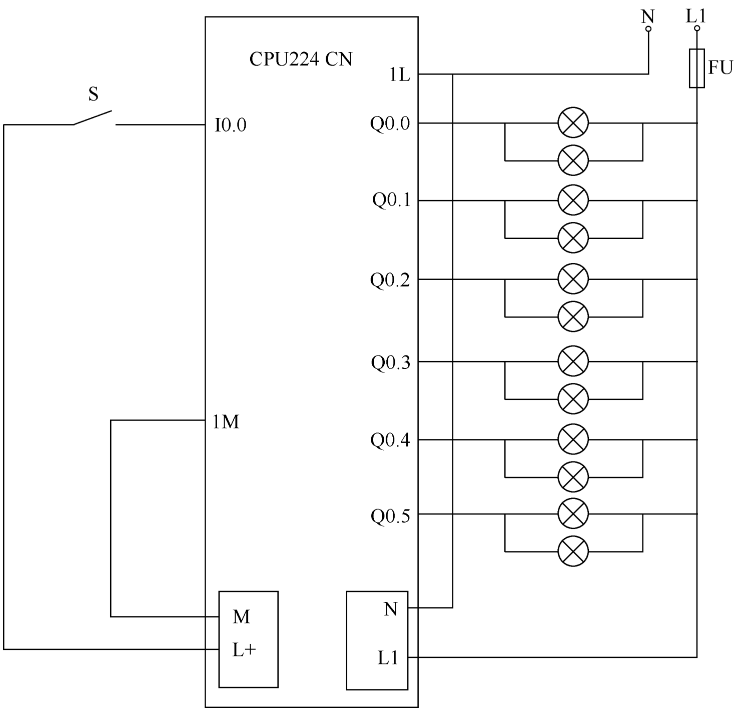


图 3. 5-1　PLC 系统接线原理图

2）梯形图程序。根据设计出的顺序功能图编写梯形图程序，如图 3. 5-3 所示。

（4）程序录入与调试

1）程序录入。根据编写的梯形图（见图 3. 5-3）录入程序。

2）运行调试。将录入的程序传送到 PLC，并进行调试，检查是否完成了控制要求，直至运行符合任务要求方为成功。

实例 2　两种液体混合装置控制系统

1. 控制要求

图 3. 5-4 为两种液体混合装置示意图。有两种待混合液体 A 和 B，对应的进料阀门为 YV1、YV2，混合液放料的阀门为 YV3。贮罐由下而上设置三个液位传感器 SL1、SL2、SL3，液面淹没时接通。

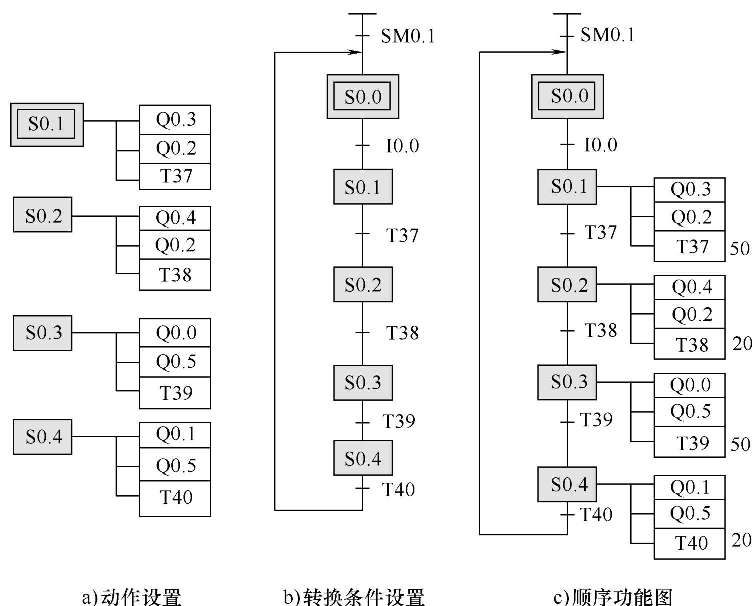


图 3. 5-2　简易红绿灯控制系统顺序功能图的设计

控制要求如下：

1）当投入运行时贮罐内为放空状态。

2）按下起动按钮 SB1 后，液体 A 的阀门 YV1 打开，液体 A 注入，当液面到达 SL2 时关闭液体 A 的阀门 YV1，打开液体 B 的阀门 YV2，当液面到达 SL3 时，关闭液体 B 的阀门

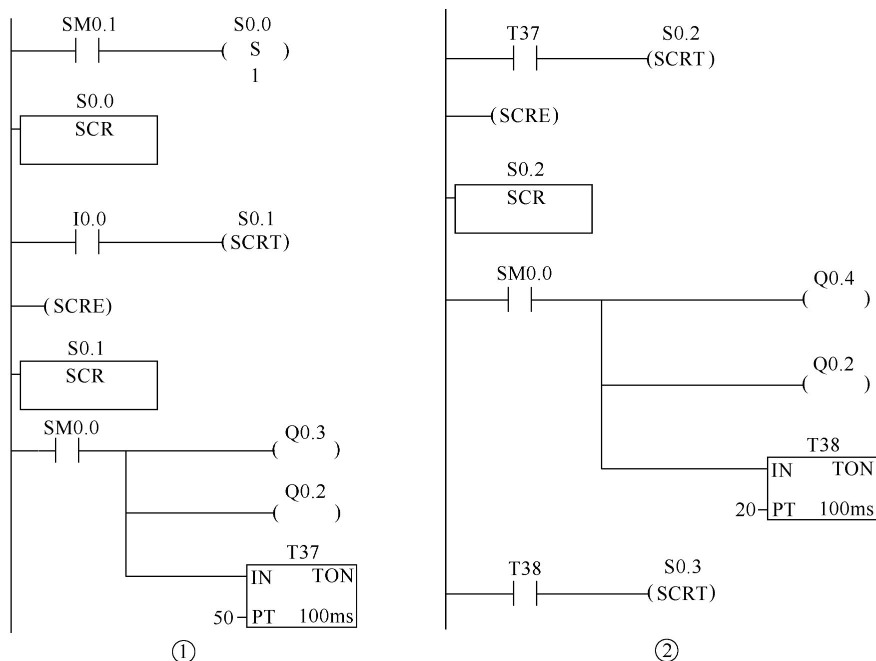


图 3. 5-3　简易红绿灯控制系统梯形图

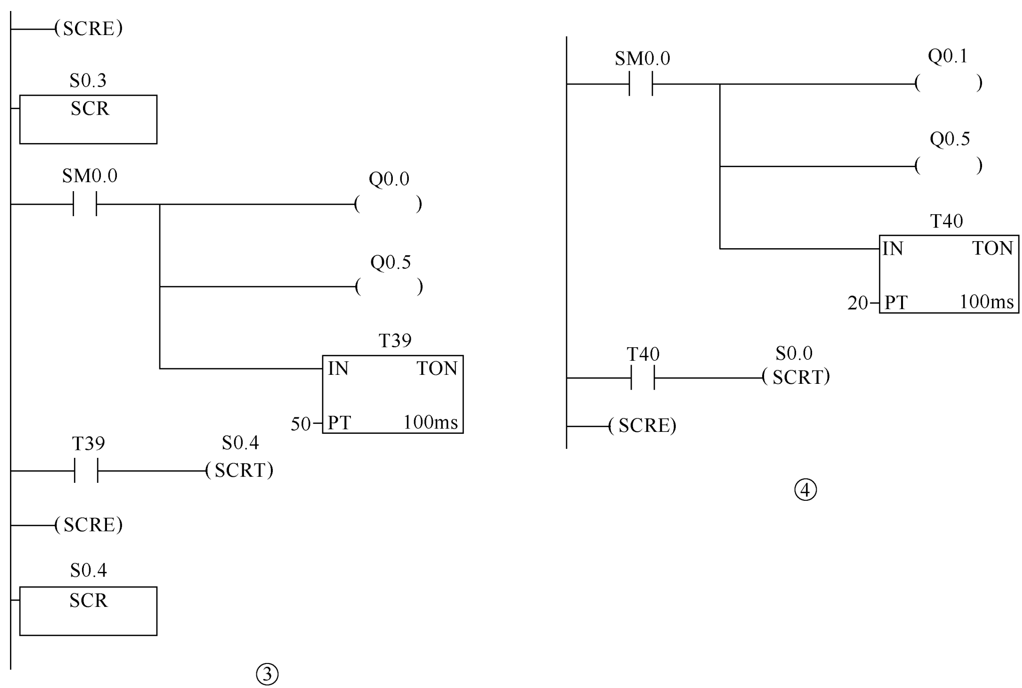


图 3. 5-3　简易红绿灯控制系统梯形图（续）

YV2，搅拌电动机开始工作，1min 后停止搅拌，阀门 YV3 打开，开始放出混合液，当液面下降到 SL1 位置时，使 SL1 由接通变为断开，再过 20s 后，贮罐放空，阀门 YV3 关闭，接着开始下一个循环。

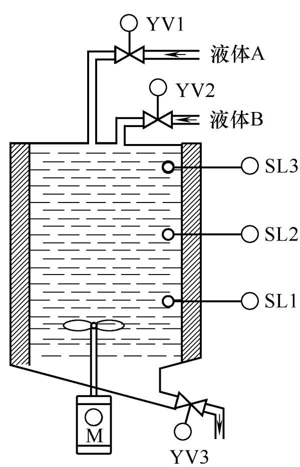


图 3. 5-4　两种液体混合装置示意图

3）按下停止按钮，要处理完当前周期的剩余工作后，系统停止在初始状态，等待下一次起动的开始。

2. 用 PLC 控制两种液体混合装置的操作步骤

（1）根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配（见表 3. 5-2）。

表 3.5-2　I/O 分配表

输入继电器	功　　能	输出继电器	功　　能
I0.0	起动按钮	Q0.0	液体 A 阀门 YV1
I0.1	停止按钮	Q0.1	液体 B 阀门 YV2
I0.2	SL1 液面传感器	Q0.2	混合液阀门 YV3
I0.3	SL2 液面传感器	Q0.3	搅拌电动机 M
I0.4	SL3 液面传感器		

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 3.5-5）

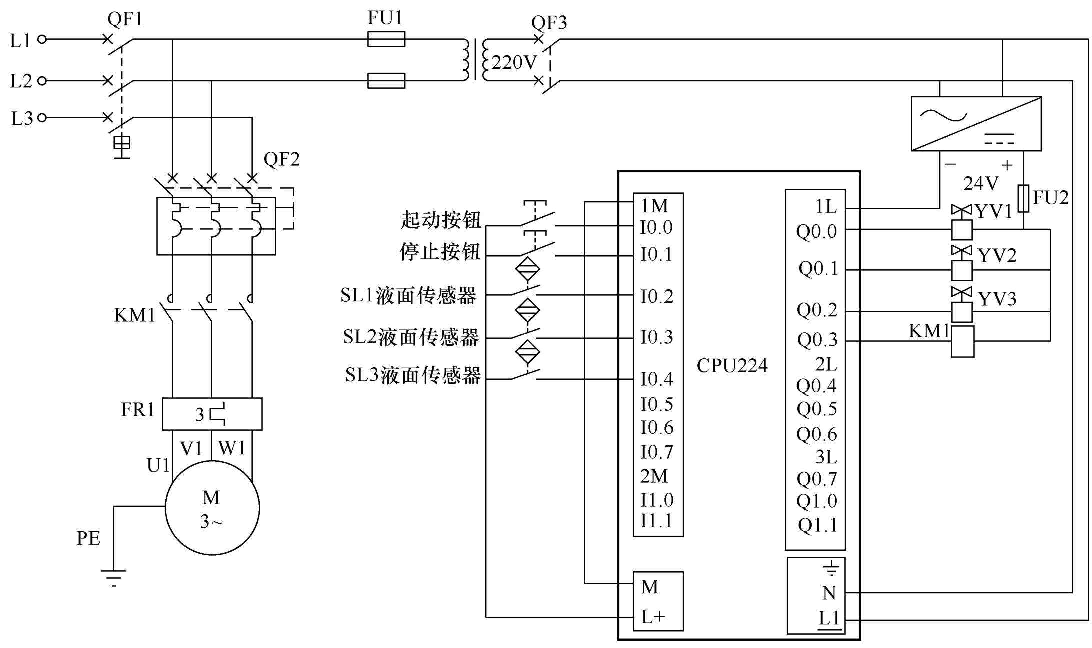


图 3.5-5　PLC 系统接线原理图

（3）程序设计

1）设计顺序功能图。图 3.5-6 为两种液体混合装置的功能流程图，它是一个典型循环结构。在实际生产的工艺流程中，若要求在某些条件下执行预定的动作，则可用跳转程序；若需要重复执行某一过程，则可用循环程序，如图 3.5-7 所示。

跳转流程：当步 2 为活动步时，若条件 f = 1，则跳过步 3 和步 4，直接激活步 5。

循环流程：当步 5 为活动步时，若条件 e = 1，则激活步 2，循环执行。

2）梯形图程序。根据功能图编写梯形图程序，如图 3.5-8 所示。

（4）程序输入与调试

把图 3.5-8 所示的程序传入 PLC 中，按照接线原理图（见图 3.5-5）配线。接通起动按钮，模拟调试程序，观察各种动作是否符合控制要求。

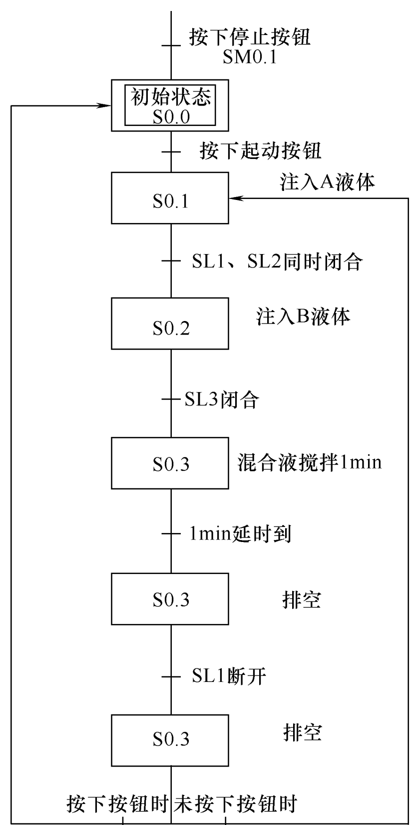


图 3.5-6　两种液体混合装置功能图

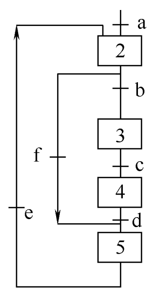


图 3.5-7　循环、跳转流程

实例 3　简易洗车控制系统

1. 控制要求

1）若方式选择开关 SA 置于 OFF 状态，当按下起动按钮 SB1 后，则按下列程序动作。

① 执行泡沫清洗；

② 按 SB3 则执行清水冲洗；

③ 按 SB4 则执行风干；

④ 按 SB5 则结束洗车。

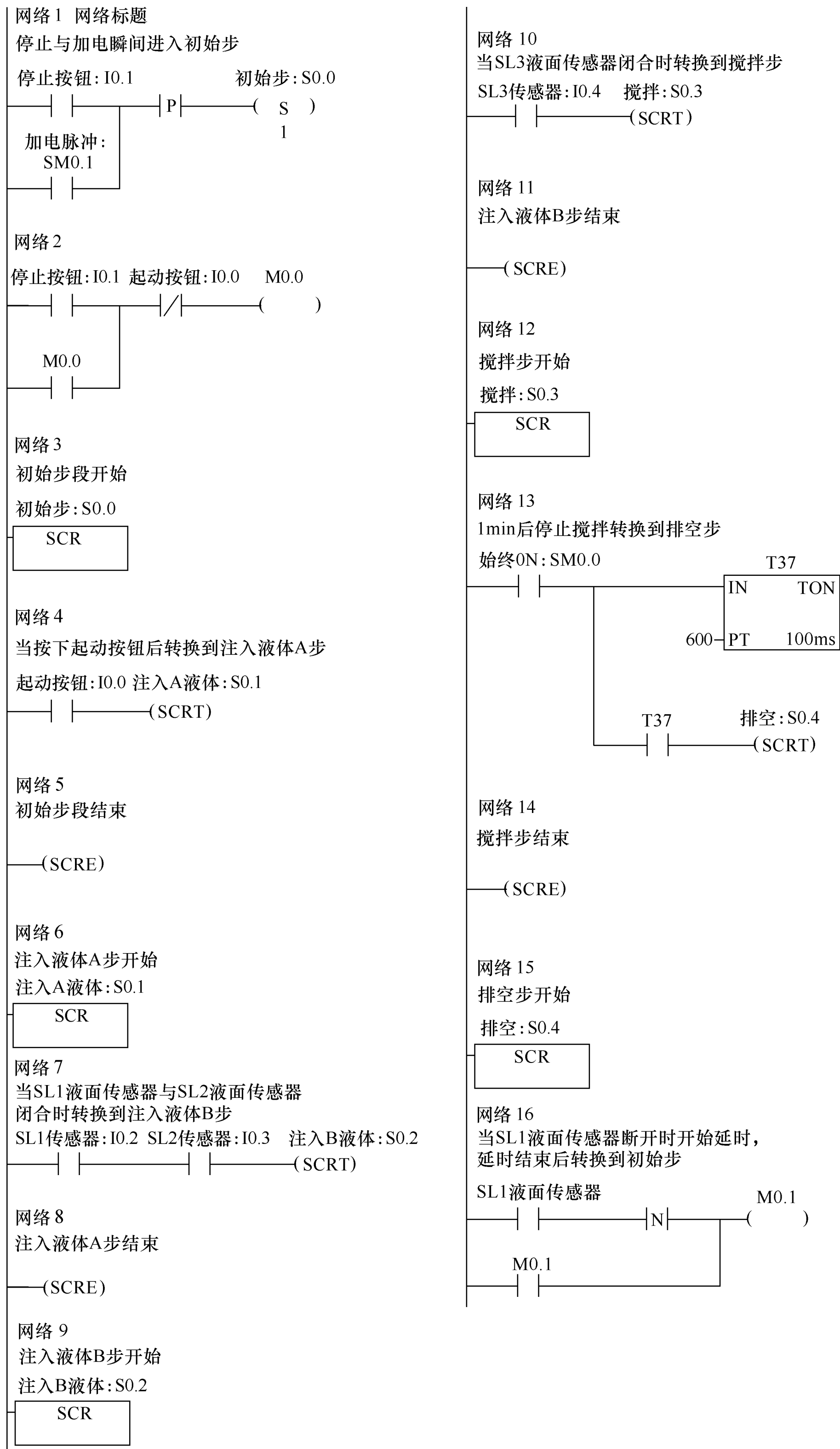


图 3.5-8 两种液体混合装置梯形图

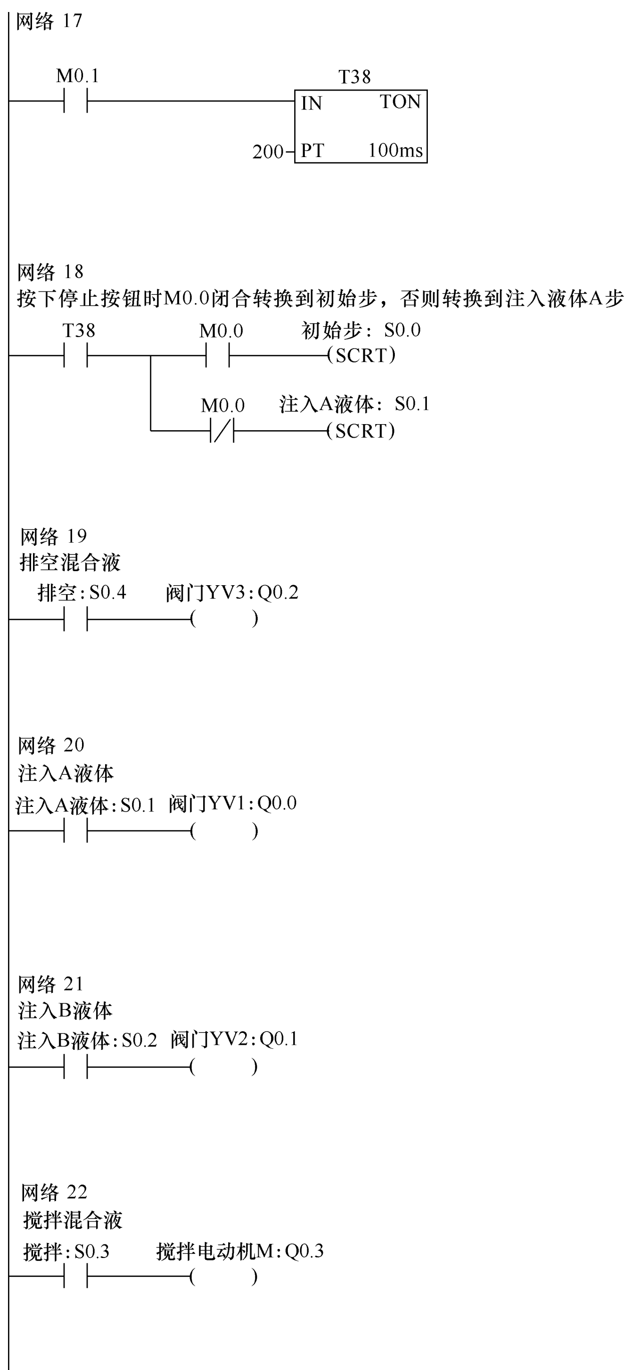


图 3.5-8　两种液体混合装置梯形图（续）

2）若方式选择开关 SA 置于 ON 状态，当按下起动按钮 SB1 后，则自动按洗车流程执行，其中泡沫清洗 10s、清水冲洗 20s、风干 5s，结束后回到待洗状态。

3）任何时候按下 SB2，则所有输出复位，停止洗车。

2. 用 PLC 控制洗车系统的操作步骤

（1）根据控制要求，分配 I/O（见表 3.5-3）

表 3.5-3　简易洗车控制系统的 I/O 分配表

输　　入		输　　出	
输　入　点	作　　用	输　出　点	作　　用
I0.0	起动按钮 SB1	Q0.0	清水清洗驱动 KM1
I0.1	方式选择开关 SA	Q0.1	泡沫清洗驱动 KM2
I0.2	停止按钮 SB2	Q0.2	风干机驱动 KM3
I0.3	手动清水冲洗按钮 SB3		
I0.4	手动风干按钮 SB4		
I0.5	手动结束按钮 SB5		

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 3.5-9）

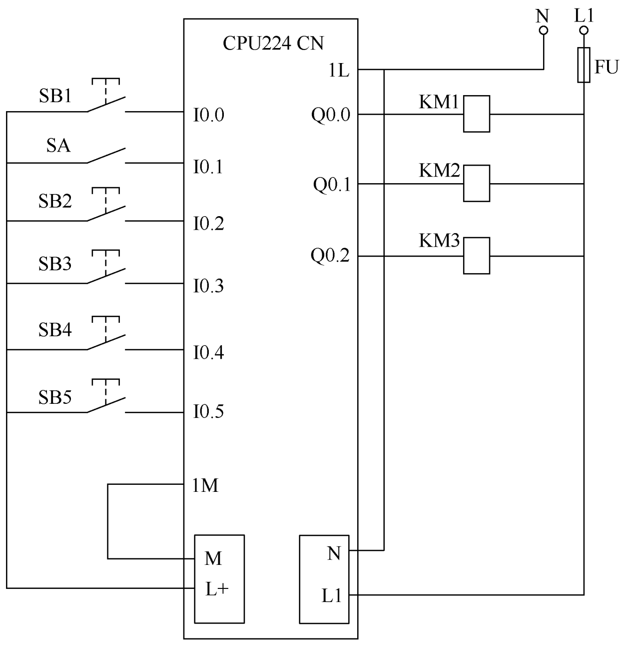


图 3.5-9　PLC 外部接线图

（3）程序设计

1）设计顺序功能图。根据转换规律和转换条件，设计顺序功能图，如图 3.5-10 所示。

2）梯形图程序。根据设计出的顺序功能图编写梯形图程序，如图 3.5-11 所示。

（4）程序录入与调试

1）程序录入。根据编写的梯形图（见图 3.5-11）录入程序。

2）运行调试。将录入的程序传送到 PLC，并进行调试，检查是否完成了控制要求，直至运行符合任务要求方为成功。

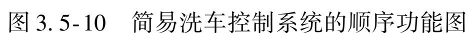


图 3. 5-10　简易洗车控制系统的顺序功能图

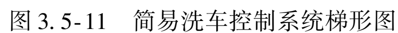


图 3. 5-11　简易洗车控制系统梯形图

实例 4　机械臂大小球分选系统

1. 控制要求

图 3.5-12 为机械臂大小球分选系统示意图。按下起动按钮 SB1 开起大小球分选系统，图中的左上角为机械原点，其动作顺序为下降→吸球→上升→右行→下降→上升→左行返回。例如，机械臂下降（设定下降时间为 2s），当电磁铁压着大球时，下限开关 SQ2 断开，压着小球时 SQ2 接通。

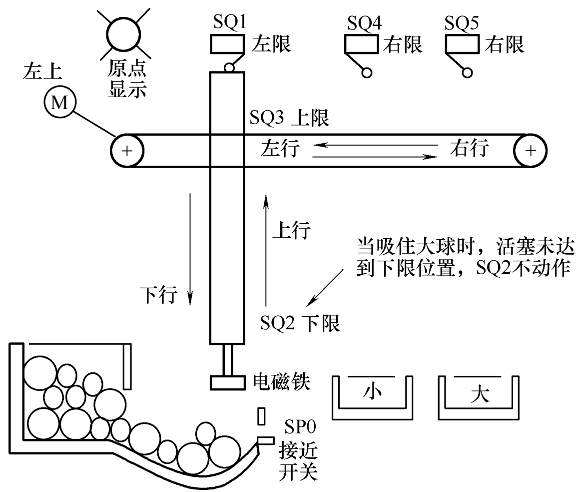


图 3.5-12　机械臂大小球分选系统示意图

2. 用 PLC 控制机械臂大小球分选系统的操作步骤

（1）根据控制要求，分配 I/O（见表 3.5-4）

表 3.5-4　I/O 分配表

输入继电器	功　　能	输出继电器	功　　能
I0.0	起动	Q0.0	下降
I0.1	左限 SQ1	Q0.1	吸盘
I0.2	下限 SQ2	Q0.2	上升
I0.3	上限 SQ3	Q0.3	右移
I0.4	右限（小球）SQ4	Q0.4	左移
I0.5	右限（大球）SQ5	Q0.5	原位显示

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 3.5-13）

（3）程序设计

1）设计顺序功能图。通过控制要求的分析，机械臂大小球分选系统功能图是一个典型的选择性分支与汇合流程结构，如图 3.5-14 所示。

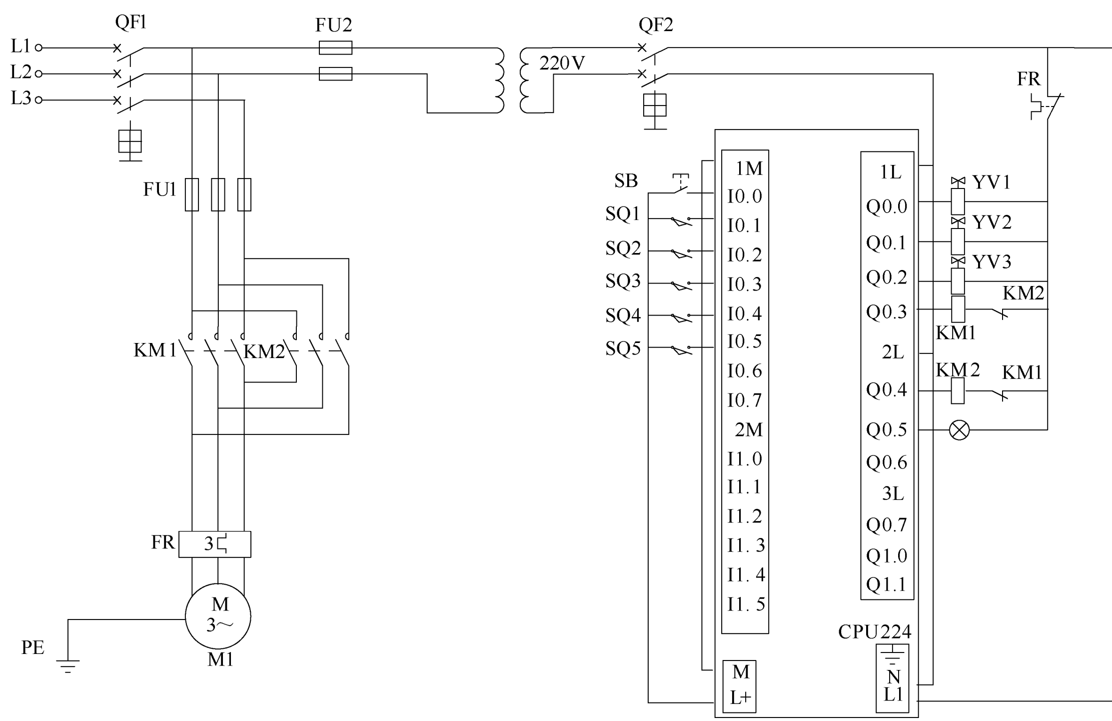


图 3. 5-13　PLC 系统接线原理图

2）梯形图程序。根据顺序功能图编写机械臂大小球分选系统的梯形图程序，如图3. 5-15所示。

（4）程序输入与调试

把图3. 5-15 所示的程序传入 PLC 中，按照原理接线图（见图3. 5-13）配线。按下起动按钮 SB 开起大小球分选系统，观察分选系统的运行是否符合控制要求。

实例5　十字路口交通灯控制系统

1. 控制要求

图3. 5-16 为十字路口交通灯的示意图。当按下起动按钮时，信号灯系统开始工作，且先南北红灯亮，东西绿灯亮；当按下停止按钮时，所有信号灯都熄灭。

南北红灯亮维持7. 5s，在南北红灯亮的同时东西绿灯也亮，并维持5s；到5s 时，东西绿灯熄灭；在东西绿灯熄灭时，东西黄灯闪亮，并维持2. 5s；到2. 5s 时，东西黄灯熄灭，东西红灯亮，同时南北红灯熄灭，绿灯亮。

东西红灯亮维持6. 5s，南北绿灯亮维持4s，然后熄灭，同时南北黄灯闪亮，维持2. 5s后熄灭，这时南北红灯亮，东西绿灯亮，周而复始。

2. 用 PLC 控制十字路口交通灯的操作步骤

（1）根据控制要求，分配 I/O（见表3. 5-5）

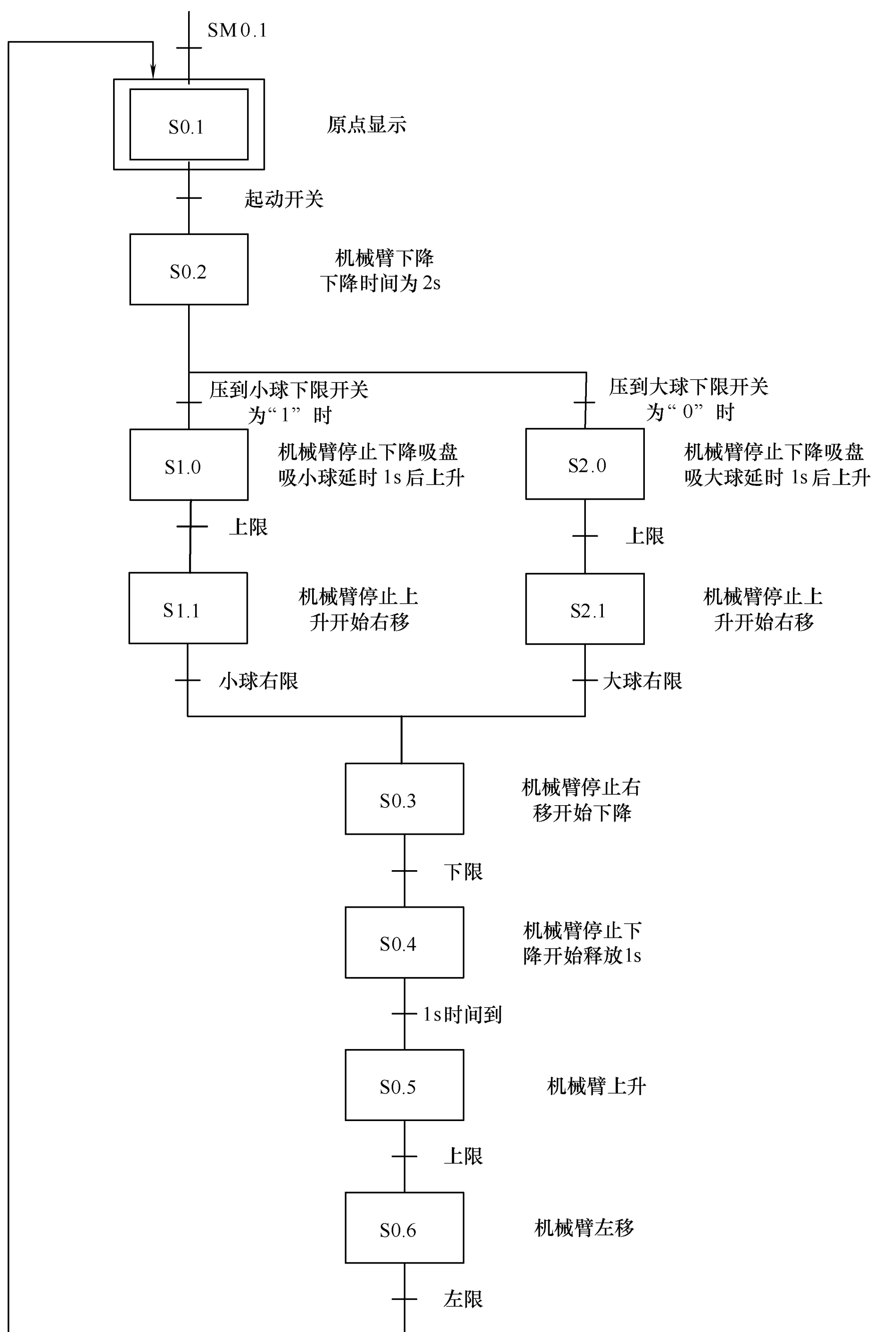


图 3.5-14　机械臂大小球分选系统功能图

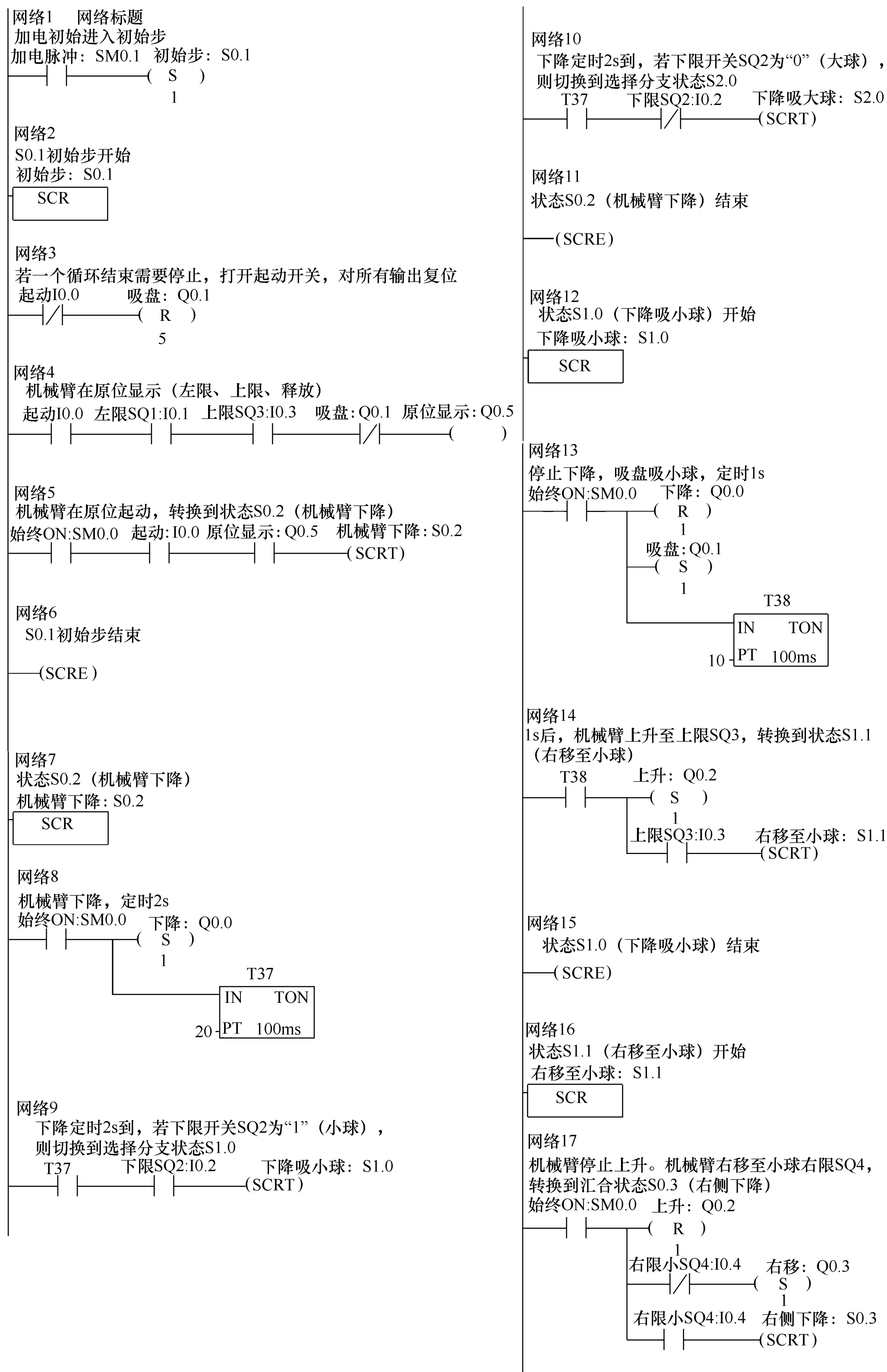


图 3. 5-15　机械臂大小球分选系统梯形图

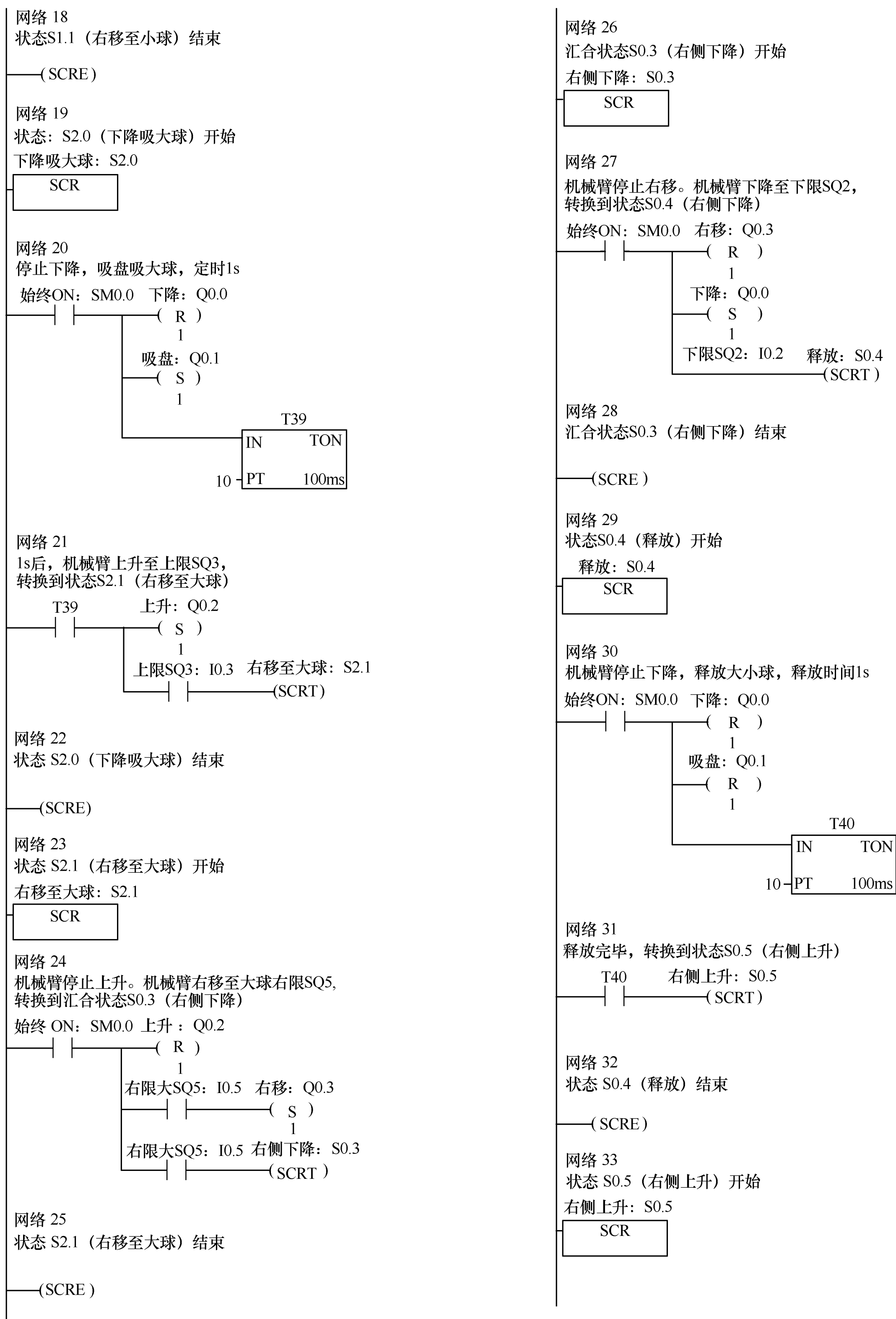
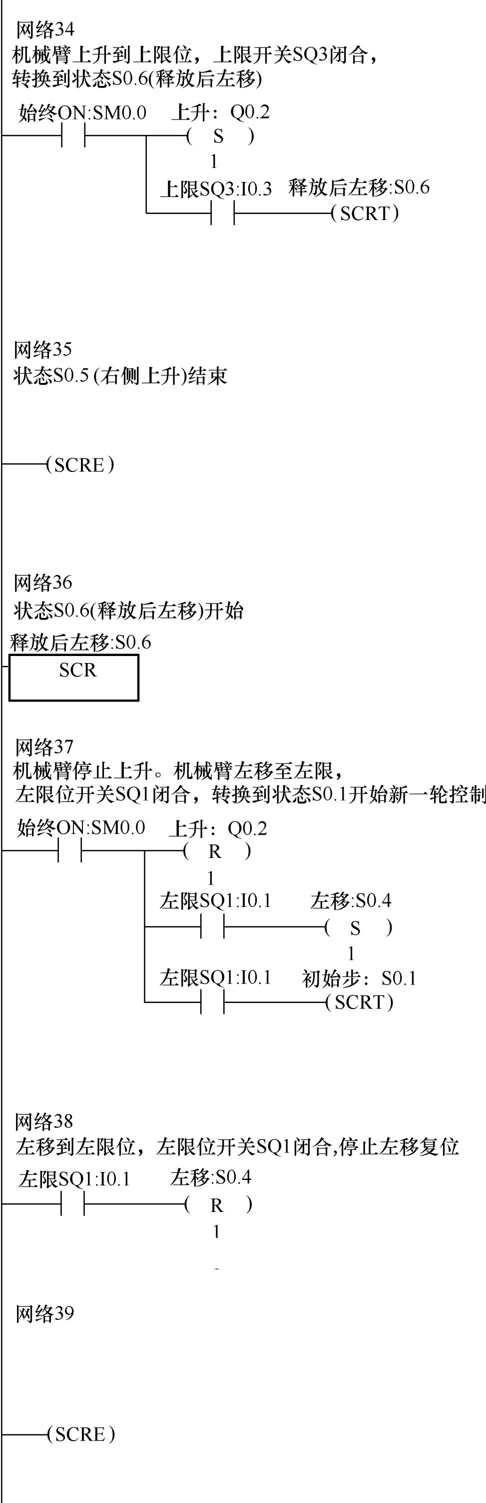


图 3.5-15　机械臂大小球



网络34
机械臂上升到上限位，上限开关SQ3闭合，
转换到状态S0.6(释放后左移)
始终ON:SM0.0
上升：Q0.2
S
1
上限SQ3:I0.3
释放后左移:S0.6
SCRT
网络35
状态S0.5 (右侧上升)结束
SCRE
网络36
状态S0.6(释放后左移)开始
释放后左移:S0.6
SCR
网络37
机械臂停止上升。机械臂左移至左限，
左限位开关SQ1闭合，转换到状态S0.1开始新一轮控制
始终ON:SM0.0
上升：Q0.2
R
1
左限SQ1:I0.1
左移:S0.4
S
1
左限SQ1:I0.1
初始步：S0.1
SCRT
网络38
左移到左限位，左限位开关SQ1闭合,停止左移复位
左限SQ1:I0.1
左移:S0.4
R
1
网络39
SCRE

分选系统梯形图（续）

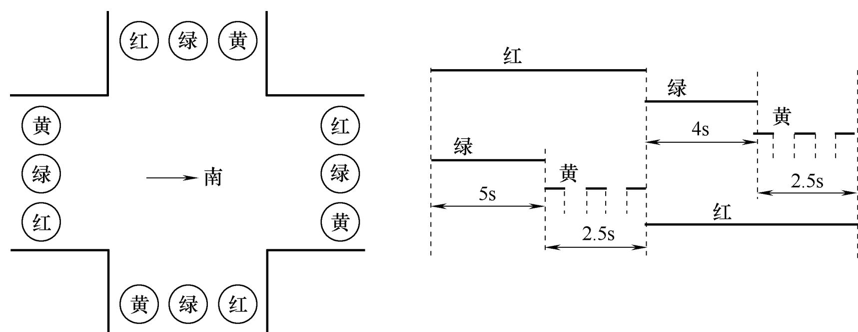


图 3.5-16　十字路口交通灯示意图

表 3.5-5　I/O 分配表

输入继电器	功　　能	输出继电器	功　　能
I0.0	起动按钮	Q0.0	南北方向红灯
I0.1	停止按钮	Q0.1	南北方向绿灯
		Q0.2	南北方向黄灯
		Q0.3	东西方向红灯
		Q0.4	东西方向绿灯
		Q0.5	东西方向黄灯

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 3.5-17）

（3）程序设计

1）设计顺序功能图。图 3.5-18 为十字路口交通灯功能流程图，它是一个典型并行分支结构。

2）梯形图程序。根据功能流程图编写梯形图程序，如图 3.5-19 所示。

（4）程序输入与调试

把图 3.5-19 所示的程序传入 PLC 中，按照接线原理图（见图 3.5-17）配线。接通起动按钮，交通信号灯进入运行状态，观察信号灯的变化是否符合控制要求。

实例 6　三层电梯的 PLC 控制

1. 控制要求

图 3.5-20 所示为采用 PLC 构成的三层简易电梯电气控制系统示意图。电梯的上、下行由 KM1 和 KM2 表示，一层有上升呼叫按钮 SB4 和指示灯 HL4，二层有上升呼叫按钮 SB6 和指示灯 HL6 以及下降呼叫按钮 SB5 和指示灯 HL5，三层有下降呼叫按钮 SB7 和指示灯 HL7。一至三层有到位限位开关 SQ1 ~ SQ3。电梯内有一至三层呼叫按钮 SB1 ~ SB3、指示灯 HL1 ~ HL3、电梯开门和关门按钮 SB10 和 SB11，电梯开门和关门分别通过 HL8 和 HL9 来表示，关门到位由限位开关 SQ5 表示，开门到位由限位开关 SQ4 表示，具体控制要求如下。

1）当电梯的轿厢停于第一层或第二层时，按第三层上升按钮，则轿厢上升至第三层后停；

2）当电梯的轿厢停于第三层或第二层时，按第一层下降按钮，则轿厢下降至第一层后停；

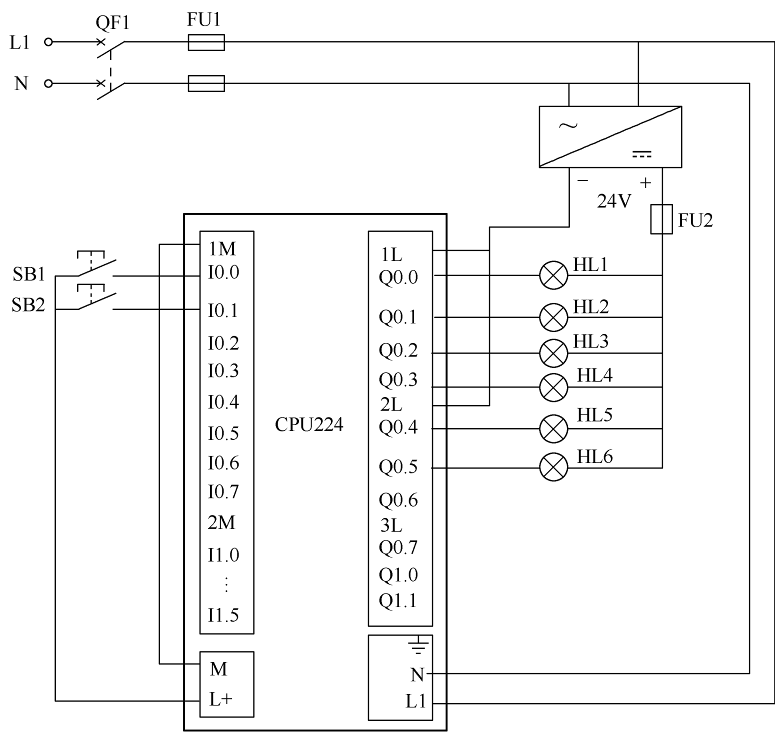


图 3. 5-17　PLC 系统接线原理图

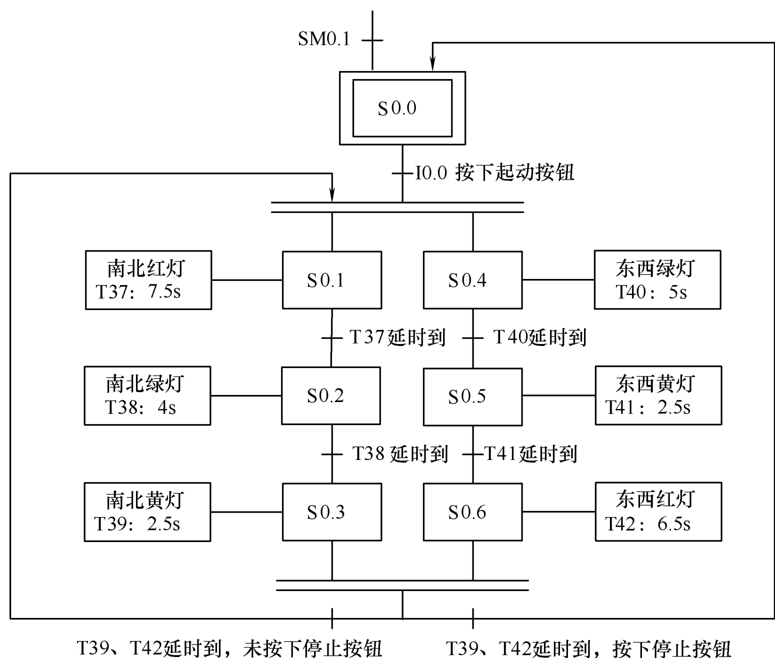


图 3. 5-18　十字路口交通灯功能图

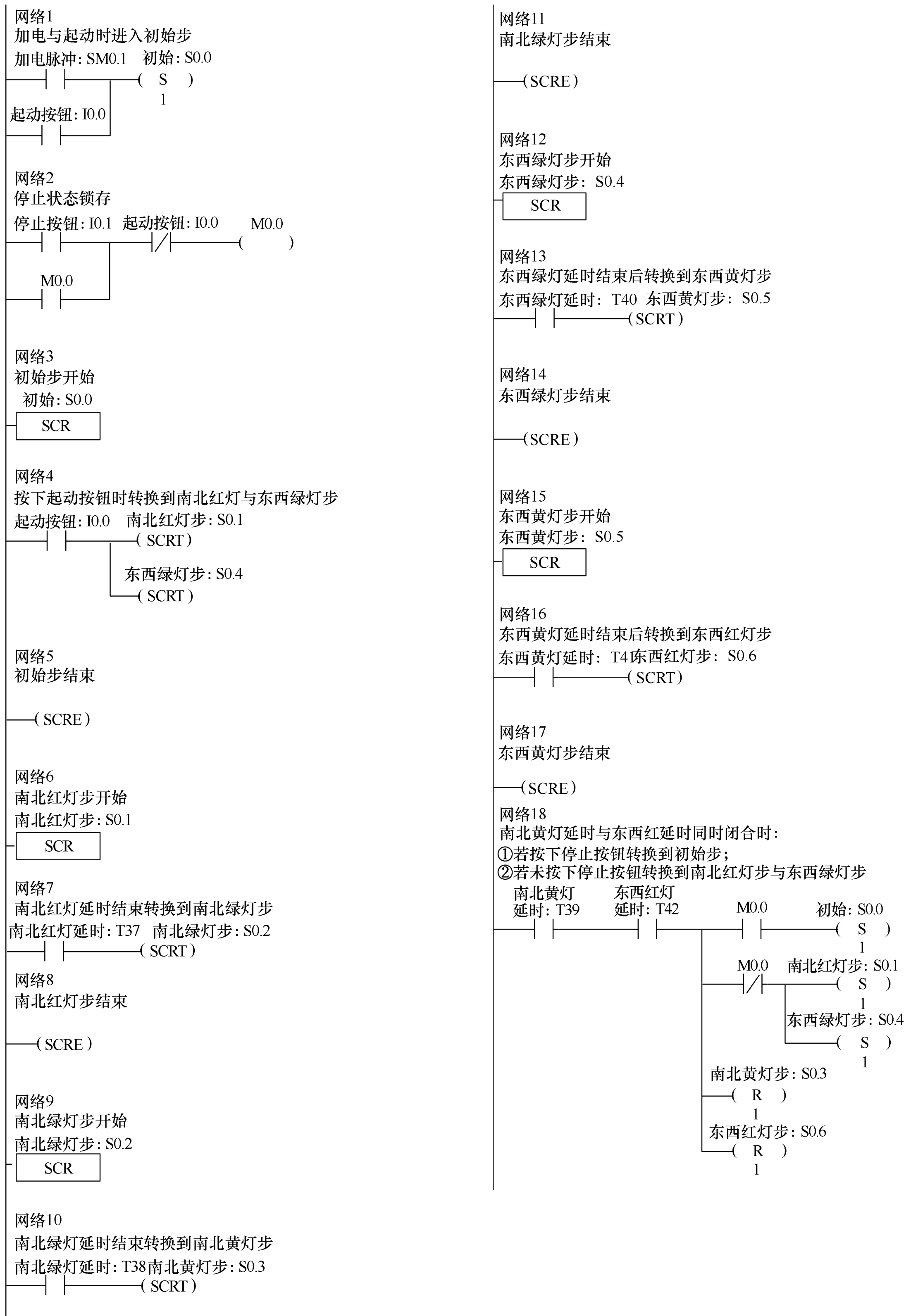
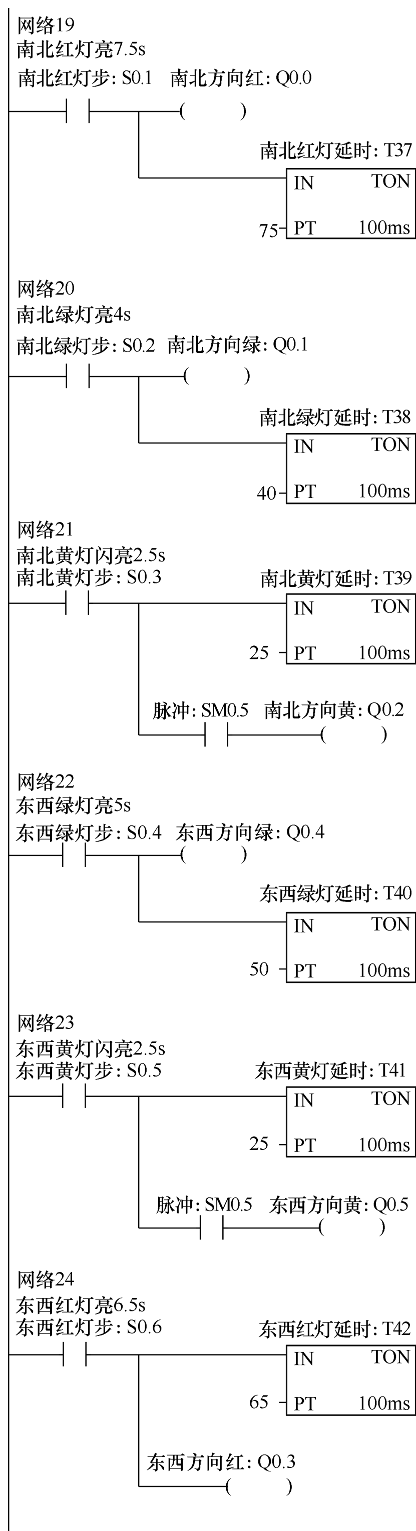

网络1
加电与起动时进入初始步
加电脉冲: SM0.1
初始: S0.0
S
1
起动按钮: I0.0
网络2
停止状态锁存
停止按钮: I0.1
起动按钮: I0.0
M0.0
M0.0
网络3
初始步开始
初始: S0.0
SCR
网络4
按下起动按钮时转换到南北红灯与东西绿灯步
起动按钮: I0.0
南北红灯步: S0.1
SCRT
东西绿灯步: S0.4
SCRT
网络5
初始步结束
SCRE
网络6
南北红灯步开始
南北红灯步: S0.1
SCR
网络7
南北红灯延时结束转换到南北绿灯步
南北红灯延时: T37
南北绿灯步: S0.2
SCRT
网络8
南北红灯步结束
SCRE
网络9
南北绿灯步开始
南北绿灯步: S0.2
SCR
网络10
南北绿灯延时结束转换到南北黄灯步
南北绿灯延时: T38
南北黄灯步: S0.3
SCRT
网络11
南北绿灯步结束
SCRE
网络12
东西绿灯步开始
东西绿灯步: S0.4
SCR
网络13
东西绿灯延时结束后转换到东西黄灯步
东西绿灯延时: T40
东西黄灯步: S0.5
SCRT
网络14
东西绿灯步结束
SCRE
网络15
东西黄灯步开始
东西黄灯步: S0.5
SCR
网络16
东西黄灯延时结束后转换到东西红灯步
东西黄灯延时: T41
东西红灯步: S0.6
SCRT
网络17
东西黄灯步结束
SCRE
网络18
南北黄灯延时与东西红延时同时闭合时:
①若按下停止按钮转换到初始步;
②若未按下停止按钮转换到南北红灯步与东西绿灯步
南北黄灯
延时: T39
东西红灯
延时: T42
M0.0
初始: S0.0
S
1
M0.0
南北红灯步: S0.1
S
1
东西绿灯步: S0.4
S
1
南北黄灯步: S0.3
R
1
东西红灯步: S0.6
R
1

图 3.5-19 十字路


网络19
南北红灯亮7.5s
南北红灯步: S0.1
南北方向红: Q0.0
南北红灯延时: T37
IN
TON
75
PT
100ms
网络20
南北绿灯亮4s
南北绿灯步: S0.2
南北方向绿: Q0.1
南北绿灯延时: T38
IN
TON
40
PT
100ms
网络21
南北黄灯闪亮2.5s
南北黄灯步: S0.3
南北黄灯延时: T39
IN
TON
25
PT
100ms
脉冲: SM0.5
南北方向黄: Q0.2
网络22
东西绿灯亮5s
东西绿灯步: S0.4
东西方向绿: Q0.4
东西绿灯延时: T40
IN
TON
50
PT
100ms
网络23
东西黄灯闪亮2.5s
东西黄灯步: S0.5
东西黄灯延时: T41
IN
TON
25
PT
100ms
脉冲: SM0.5
东西方向黄: Q0.5
网络24
东西红灯亮6.5s
东西红灯步: S0.6
东西红灯延时: T42
IN
TON
65
PT
100ms
东西方向红: Q0.3

口交通灯梯形图

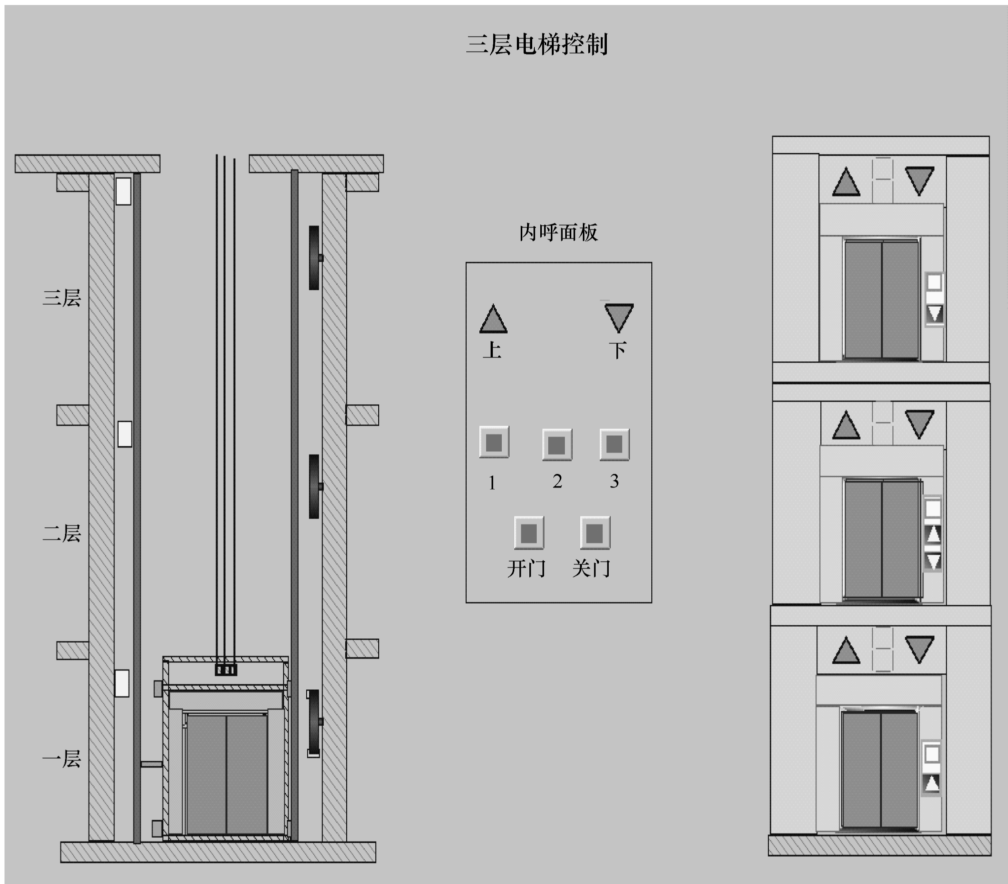


图 3. 5-20　三层电梯的 PLC 控制示意图

3）当轿厢停在第一层时，按第二层上升按钮，则轿厢上升至第二层后停；

4）当轿厢停在第三层时，按第二层下降按钮，则轿厢下降至第二层后停；

5）当轿厢停在第一层时，若第二层、第三层均有呼梯信号，则轿厢上升至第二层暂停后，继续上升至第三层；

6）当轿厢停在第三层时，若第二层、第一层均有呼梯信号，则轿厢下降至第二层暂停后，继续下降至第一层。

2. 用 PLC 控制三层电梯

（1）根据控制要求，分配 I/O（见表 3. 5-6）

表 3. 5-6　I/O 分配表

输入继电器	功　　能	输出继电器	功　　能
I0. 1	SB1 电梯内一层按钮	Q0. 1	HL1 电梯一层指示灯
I0. 2	SB2 电梯内二层按钮	Q0. 2	HL2 电梯二层指示灯
I0. 3	SB3 电梯内三层按钮	Q0. 3	HL3 电梯三层指示灯
I0. 4	SB4 一层上升呼叫按钮	Q0. 0	KM3 电梯开门
I0. 5	SB5 二层下降呼叫按钮	Q0. 4	KM4 电梯关门

（续）

输入继电器	功　　能	输出继电器	功　　能
I0.6	SB6 二层上升呼叫按钮	Q0.5	KM1 轿厢上升
I0.7	SB7 三层下降呼叫按钮	Q0.6	KM2 轿厢下降
I1.0	SQ1 电梯一层到位限位开关		
I1.1	SQ2 电梯二层到位限位开关		
I1.2	SQ3 电梯三层到位限位开关		
I0.0	SB10 电梯开门按钮		
I1.5	SB11 电梯关门按钮		
I1.3	SQ4 电梯开门检测		
I1.4	SQ5 电梯关门检测		

（2）根据控制要求，设计并绘制 PLC 系统接线原理图（见图 3.5-21）

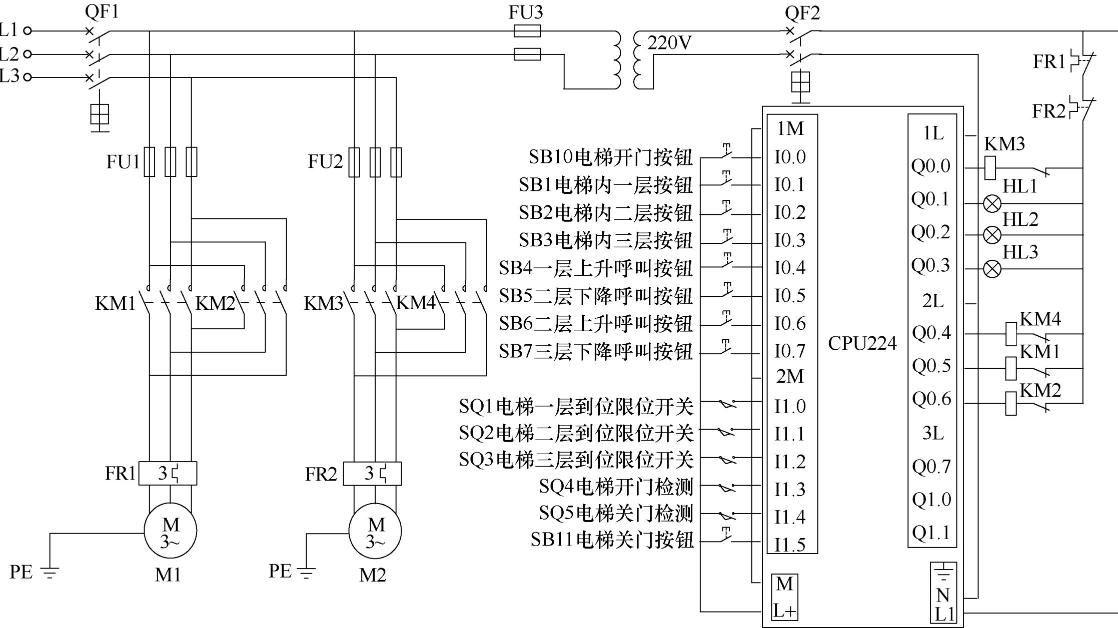


图 3.5-21　PLC 系统接线原理图

（3）程序设计

1）设计顺序功能图。电梯是根据外部呼叫信号以及自身控制规律等运行的，而呼叫是随机的，所以电梯控制系统实际上是一个人机交互式的控制系统，单纯用顺序控制或逻辑控制是不能满足控制要求的。因此，在本实例中将电梯控制程序划分为“数据采集与显示”、“呼梯判断”、“上行处理”及“下行处理”几个子程序。“数据采集与显示”采用逻辑控制，“呼梯判断”、“上行处理”及“下行处理”采用顺序控制，总体设计如图 3.5-22 所示。

呼梯判断由顺序控制流程完成，主要依据是内呼梯优先，由先判断内呼来保证这一点，呼梯判断功能图如图 3.5-23 所示。

上、下行处理由顺序控制流程完成，在上、下行流程中，遵守的主要原则是“内呼优

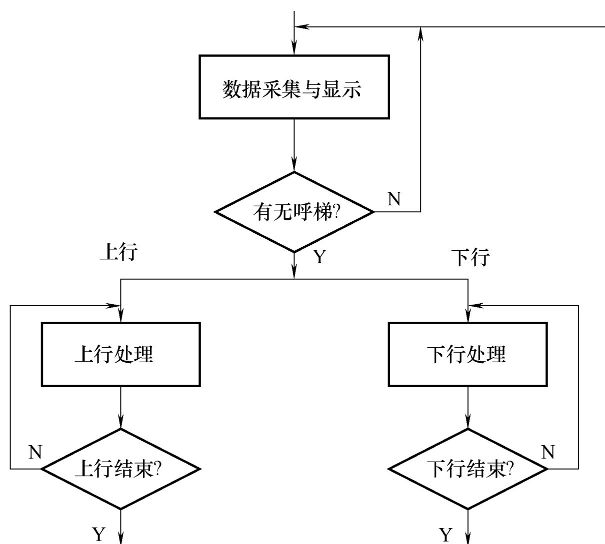


图 3.5-22　主程序流程图

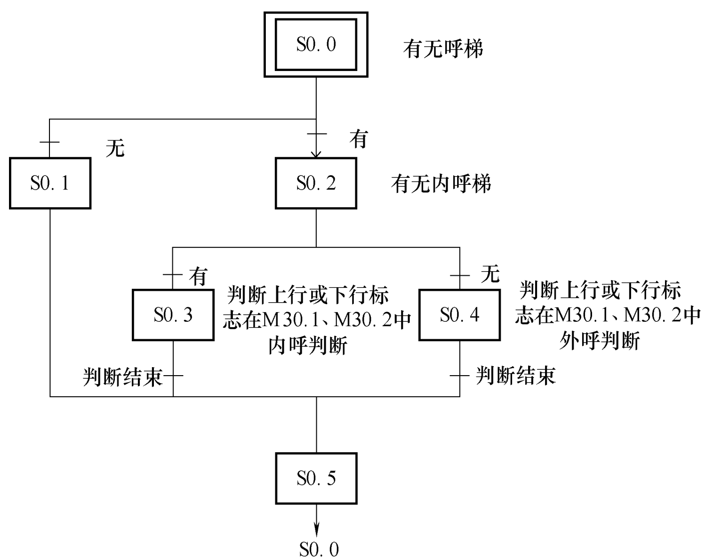


图 3.5-23　呼梯判断功能图

先，上行上呼优先，下行下呼优先”，上行流程与下行流程相对称，因此只列出上行处理功能图，如图 3.5-24 所示。

2）梯形图程序。根据顺序功能图编写三层简易电梯电气控制系统的梯形图程序，如图 3.5-25所示。

（4）程序输入与调试　把图 3.5-25 所示的程序传入 PLC 中，按照原理接线图（见图 3.5-21）配线。接通相应平层限位开关，分别将电梯置于第一层至第三层；分别按下内呼梯按钮、外呼梯按钮，观察电梯的升降运行是否符合控制要求；按下开门、关门按钮观察电梯的门控动作是否正常。

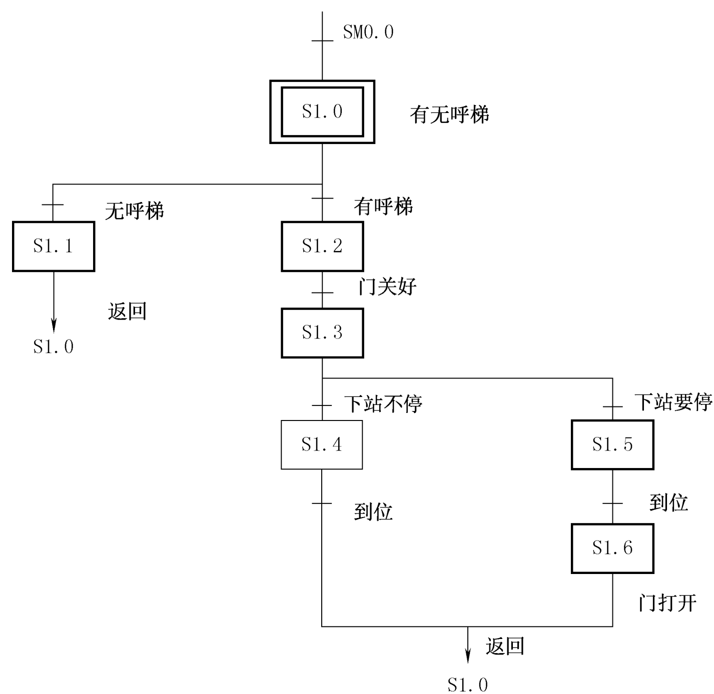


图 3.5-24　上行处理功能图



图 3.5-25　三层电梯的 PLC 控制梯形图

子程序1：数据采集与显示

网络 1

加电脉冲：SM0.1　定向向上：M30.1 (R) 1

定向向下：M30.2 (R) 1

网络 2　楼层信号采集

加电脉冲：SM0.1

平层1层：I1.0

MOV_B　EN　ENO　1 - IN　OUT - 楼层号：VB0

网络 3

平层2层：I1.1

MOV_B　EN　ENO　2 - IN　OUT - 楼层号：VB0

网络 4

平层3层：I1.2

MOV_B　EN　ENO　3 - IN　OUT - 楼层号：VB0

网络 5　楼层显示

始终闭合：SM0.0　平层1层：I1.0　楼层指示1：Q0.1 (S) 1

楼层指示2：Q0.2 (R) 1

楼层指示3：Q0.3 (R) 1

平层2层：I1.1　楼层指示2：Q0.2 (S) 1

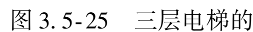
楼层指示1：Q0.1 (R) 1

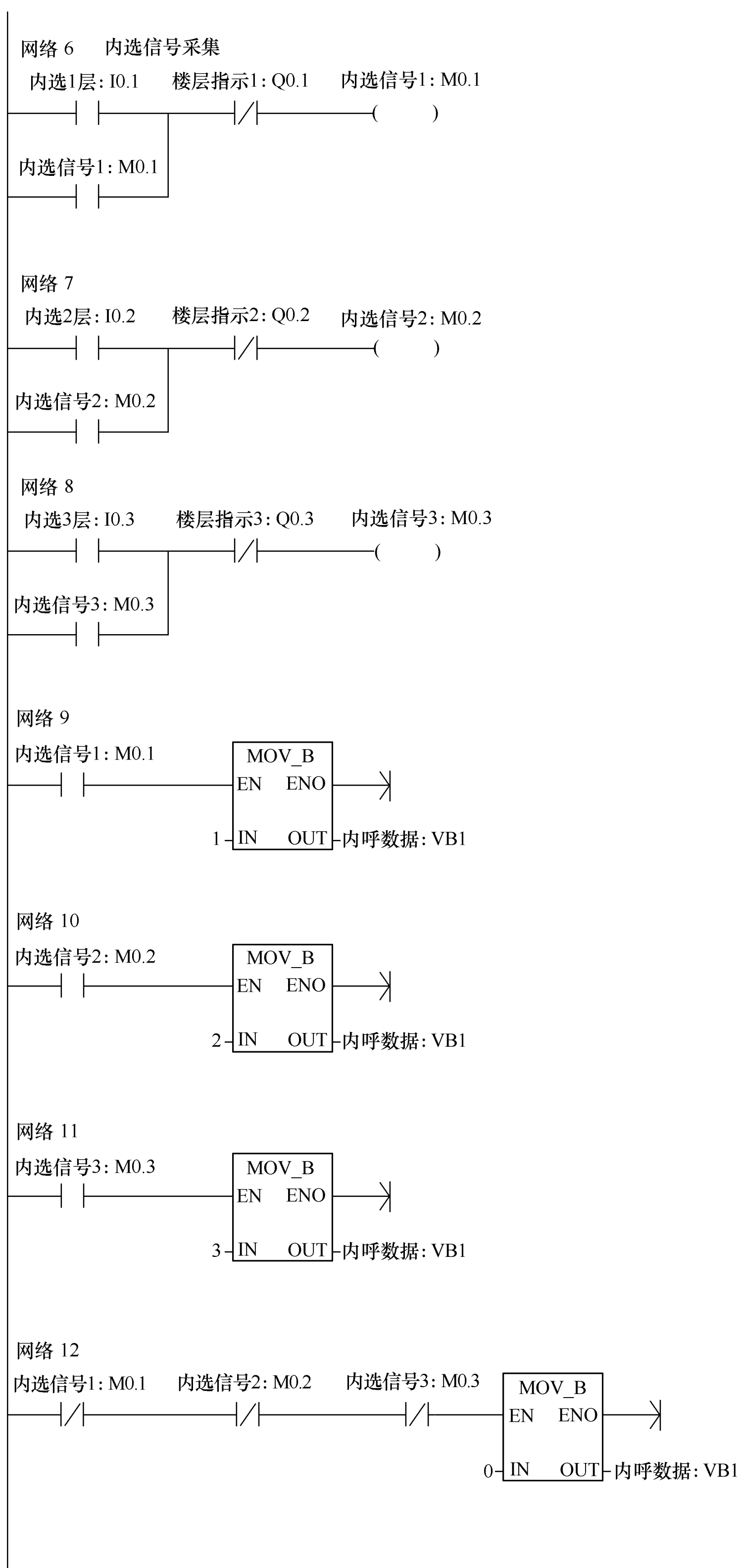
楼层指示3：Q0.3 (R) 1

平层3层：I1.2　楼层指示3：Q0.3 (S) 1

楼层指示1：Q0.1 (R) 1

楼层指示2：Q0.2 (R) 1


图 3.5-25　三层电梯的



网络 6　内选信号采集
内选1层: I0.1
楼层指示1: Q0.1
内选信号1: M0.1
内选信号1: M0.1
网络 7
内选2层: I0.2
楼层指示2: Q0.2
内选信号2: M0.2
内选信号2: M0.2
网络 8
内选3层: I0.3
楼层指示3: Q0.3
内选信号3: M0.3
内选信号3: M0.3
网络 9
内选信号1: M0.1
MOV_B
EN ENO
1 IN OUT 内呼数据: VB1
网络 10
内选信号2: M0.2
MOV_B
EN ENO
2 IN OUT 内呼数据: VB1
网络 11
内选信号3: M0.3
MOV_B
EN ENO
3 IN OUT 内呼数据: VB1
网络 12
内选信号1: M0.1
内选信号2: M0.2
内选信号3: M0.3
MOV_B
EN ENO
0 IN OUT 内呼数据: VB1

PLC 控制梯形图（续）

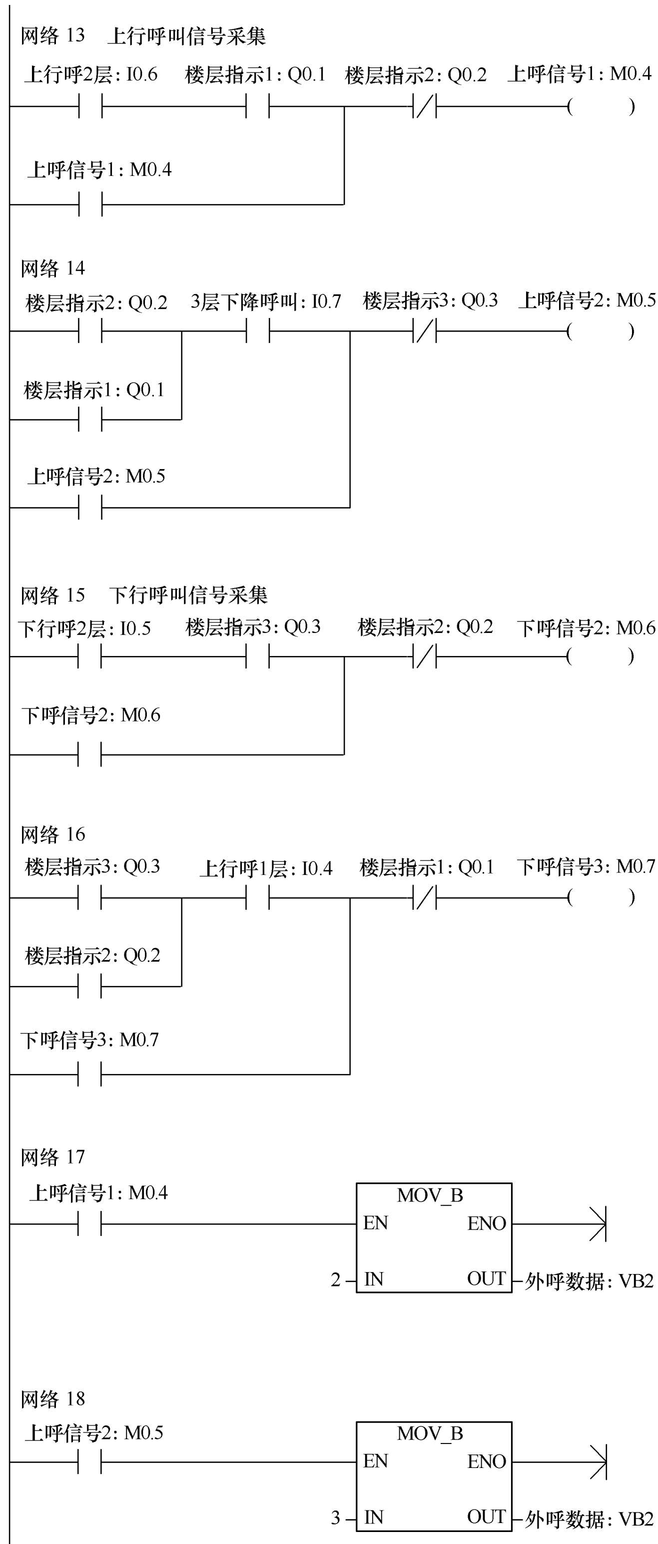
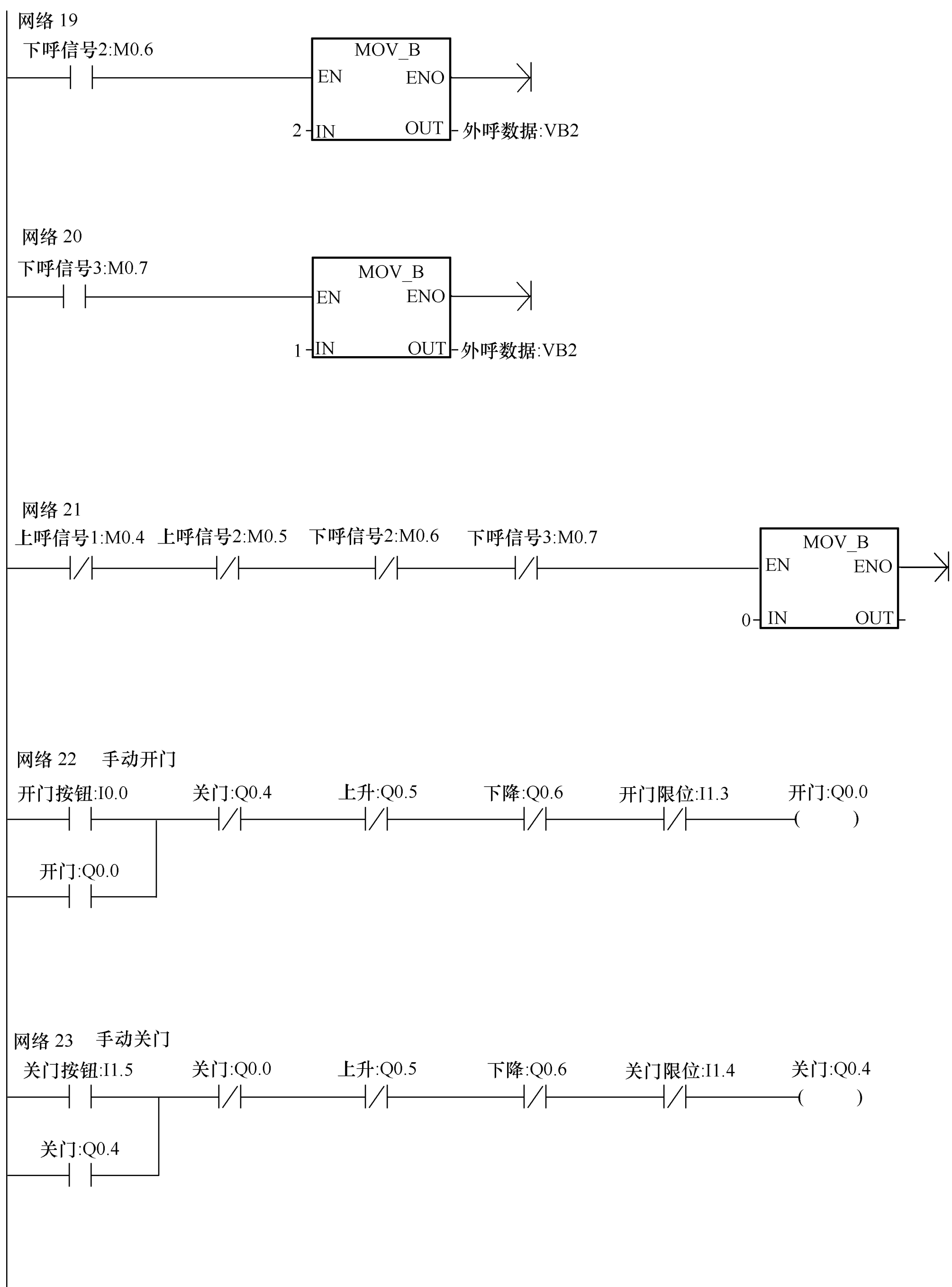

网络 13　上行呼叫信号采集
上行呼2层: I0.6
楼层指示1: Q0.1
楼层指示2: Q0.2
上呼信号1: M0.4
上呼信号1: M0.4
网络 14
楼层指示2: Q0.2
3层下降呼叫: I0.7
楼层指示3: Q0.3
上呼信号2: M0.5
楼层指示1: Q0.1
上呼信号2: M0.5
网络 15　下行呼叫信号采集
下行呼2层: I0.5
楼层指示3: Q0.3
楼层指示2: Q0.2
下呼信号2: M0.6
下呼信号2: M0.6
网络 16
楼层指示3: Q0.3
上行呼1层: I0.4
楼层指示1: Q0.1
下呼信号3: M0.7
楼层指示2: Q0.2
下呼信号3: M0.7
网络 17
上呼信号1: M0.4
MOV_B
EN
ENO
2
IN
OUT
外呼数据: VB2
网络 18
上呼信号2: M0.5
MOV_B
EN
ENO
3
IN
OUT
外呼数据: VB2

图 3. 5-25　三层电梯的


网络 19
下呼信号2:M0.6
MOV_B
EN
ENO
2
IN
OUT
外呼数据:VB2
网络 20
下呼信号3:M0.7
MOV_B
EN
ENO
1
IN
OUT
外呼数据:VB2
网络 21
上呼信号1:M0.4
上呼信号2:M0.5
下呼信号2:M0.6
下呼信号3:M0.7
MOV_B
EN
ENO
0
IN
OUT
网络 22 手动开门
开门按钮:I0.0
关门:Q0.4
上升:Q0.5
下降:Q0.6
开门限位:I1.3
开门:Q0.0
开门:Q0.0
网络 23 手动关门
关门按钮:I1.5
关门:Q0.0
上升:Q0.5
下降:Q0.6
关门限位:I1.4
关门:Q0.4
关门:Q0.4

PLC 控制梯形图（续）

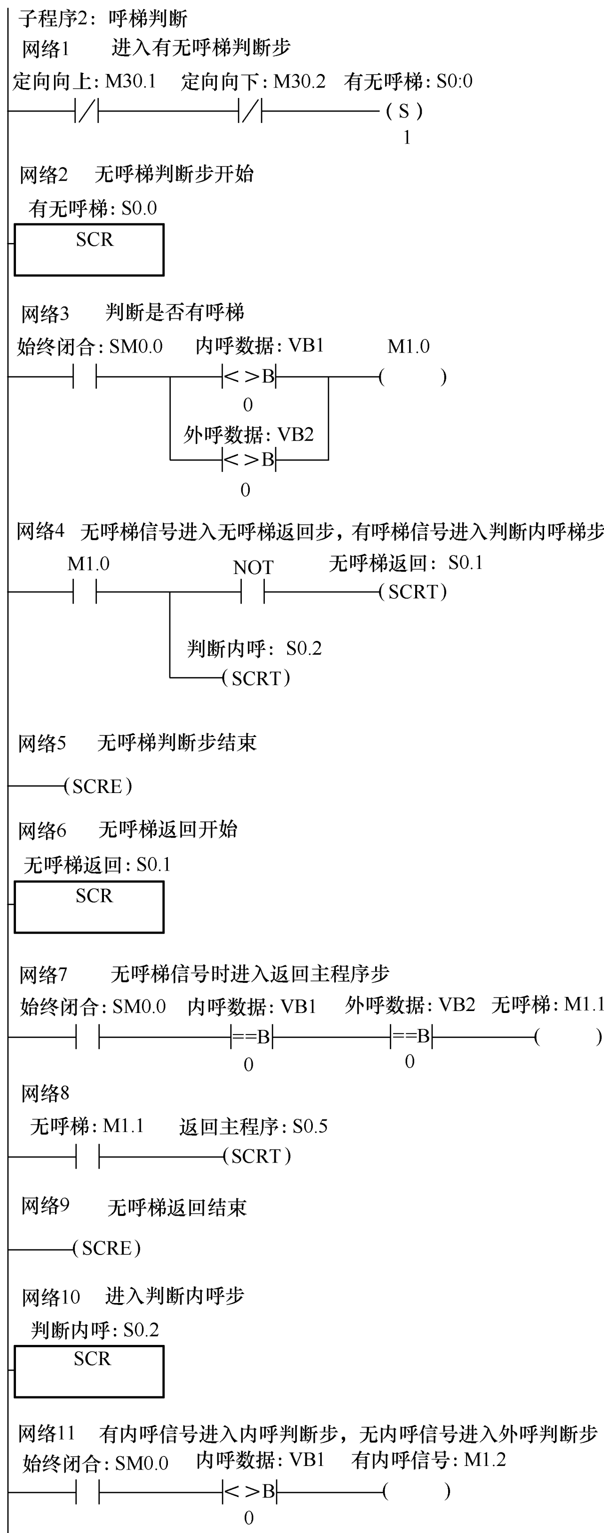

子程序2：呼梯判断
网络1 进入有无呼梯判断步
定向向上: M30.1
定向向下: M30.2
有无呼梯: S0:0
S
1
网络2 无呼梯判断步开始
有无呼梯: S0.0
SCR
网络3 判断是否有呼梯
始终闭合: SM0.0
内呼数据: VB1
< >B
0
M1.0
外呼数据: VB2
< >B
0
网络4 无呼梯信号进入无呼梯返回步，有呼梯信号进入判断内呼梯步
M1.0
NOT
无呼梯返回: S0.1
SCRT
判断内呼: S0.2
SCRT
网络5 无呼梯判断步结束
SCRE
网络6 无呼梯返回开始
无呼梯返回: S0.1
SCR
网络7 无呼梯信号时进入返回主程序步
始终闭合: SM0.0
内呼数据: VB1
==B
0
外呼数据: VB2
==B
0
无呼梯: M1.1
网络8
无呼梯: M1.1
返回主程序: S0.5
SCRT
网络9 无呼梯返回结束
SCRE
网络10 进入判断内呼步
判断内呼: S0.2
SCR
网络11 有内呼信号进入内呼判断步，无内呼信号进入外呼判断步
始终闭合: SM0.0
内呼数据: VB1
< >B
0
有内呼信号: M1.2

图 3.5-25 三层电梯的

网络12

有内呼信号：M1.2　NOT　外呼判断：S0.4
(SCRT)

内呼判断：S0.3
(SCRT)

网络13　判断内呼步结束

(SCRE)

网络14　内呼判断步开始

内呼判断：S0.3
SCR

网络15　内呼判断：呼叫信号大于平层信号定向向上M30.1闭合
呼叫信号小于平层信号定向向下M30.2闭合

始终闭合：SM0.0　内呼数据：VB1　定向向上：M30.1
>B　(S)
1
楼层号：VB0
定向向下：M30.2
(R)
1
内呼数据：VB1　定向向下：M30.2
<B　(S)
1
楼层号：VB0
定向向上：M30.1
(R)
1

网络16

定向向上：M30.1　返回主程序：S0.5
(SCRT)
定向向下：M30.2

网络17　内呼判断步结束

(SCRE)

网络18　外呼判断步开始

外呼判断：S0.4
SCR

网络19　外呼判断：呼叫信号大于平层信号定向向上M30.1闭合
呼叫信号小于平层信号定向向下M30.2闭合

始终闭合：SM0.0　外呼数据：VB2　定向向上：M30.1
>B　(S)
1
楼层号：VB0
定向向下：M30.2
(R)
1
外呼数据：VB2　定向向下：M30.2
<B　(S)
1
楼层号：VB0
定向向上：M30.1
(R)
1


PLC 控制梯形图（续）

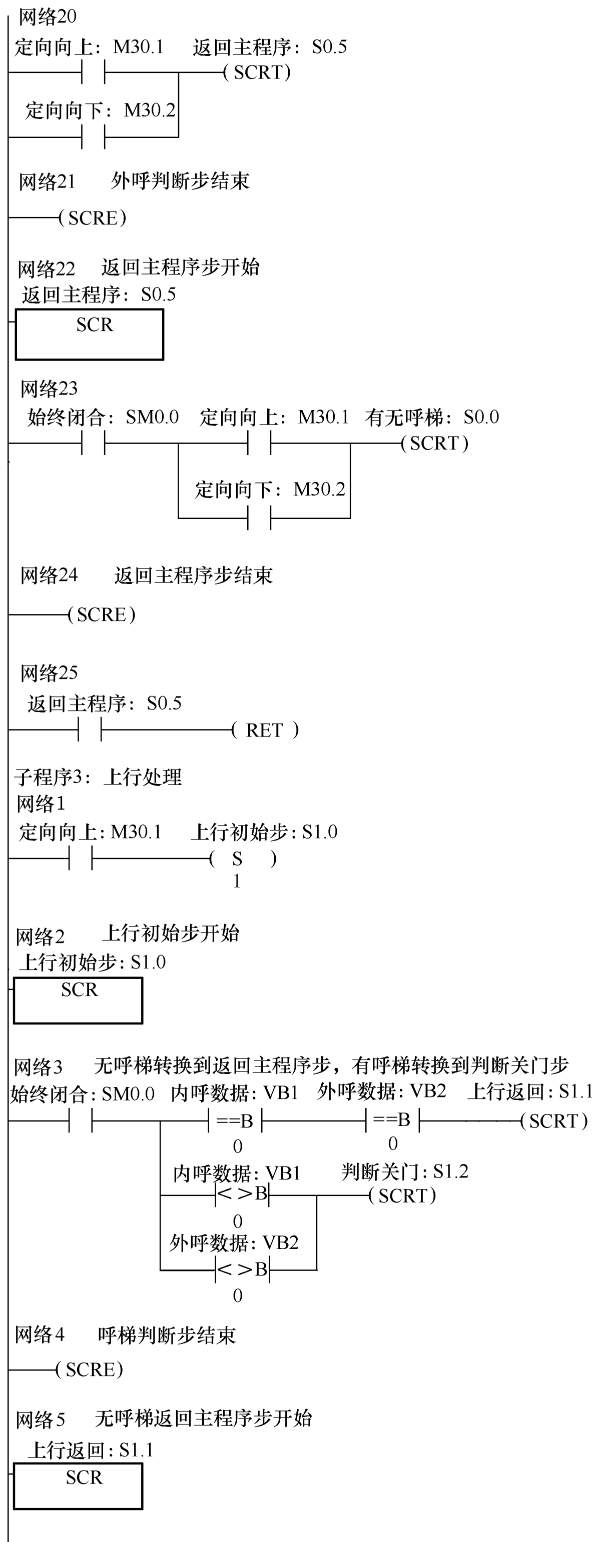
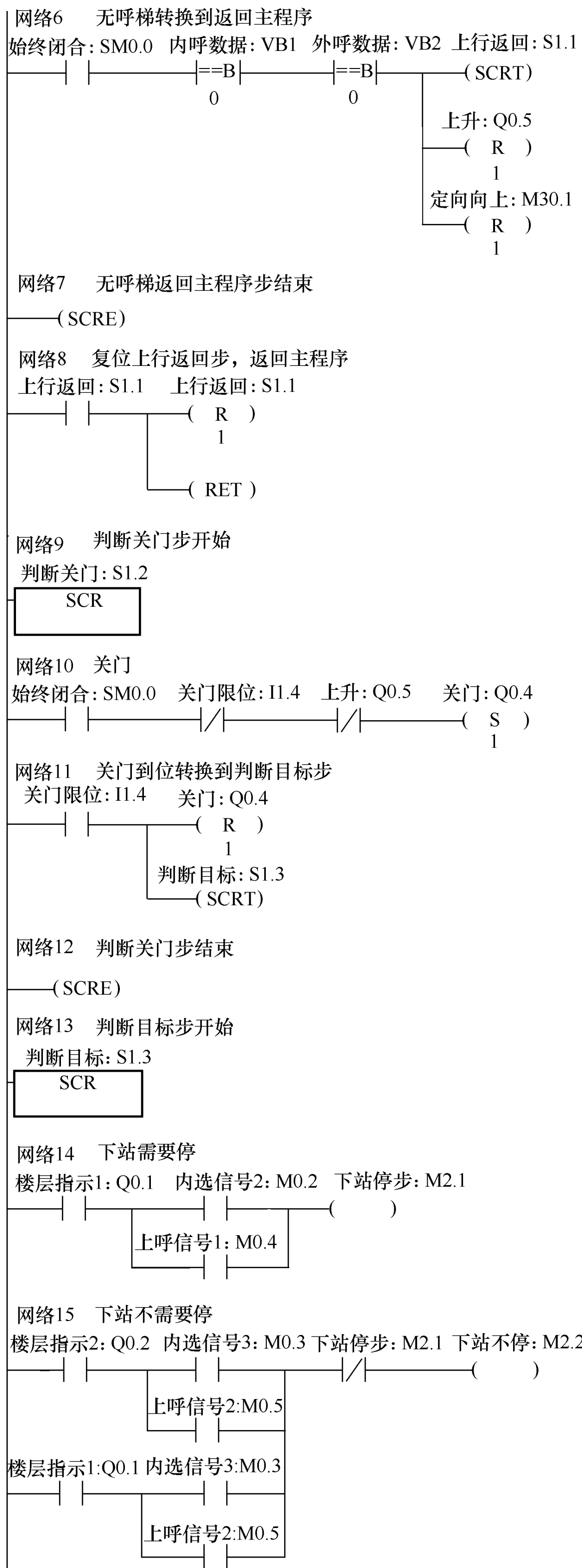

网络20
定向向上：M30.1
返回主程序：S0.5
(SCRT)
定向向下：M30.2
网络21 外呼判断步结束
(SCRE)
网络22 返回主程序步开始
返回主程序：S0.5
SCR
网络23
始终闭合：SM0.0
定向向上：M30.1
有无呼梯：S0.0
(SCRT)
定向向下：M30.2
网络24 返回主程序步结束
(SCRE)
网络25
返回主程序：S0.5
(RET)
子程序3：上行处理
网络1
定向向上：M30.1
上行初始步：S1.0
(S)
1
网络2 上行初始步开始
上行初始步：S1.0
SCR
网络3 无呼梯转换到返回主程序步，有呼梯转换到判断关门步
始终闭合：SM0.0
内呼数据：VB1
外呼数据：VB2
上行返回：S1.1
==B
0
==B
0
(SCRT)
内呼数据：VB1
判断关门：S1.2
< >B
0
(SCRT)
外呼数据：VB2
< >B
0
网络4 呼梯判断步结束
(SCRE)
网络5 无呼梯返回主程序步开始
上行返回：S1.1
SCR

图 3.5-25 三层电梯的


网络6　无呼梯转换到返回主程序
始终闭合：SM0.0
内呼数据：VB1
外呼数据：VB2
上行返回：S1.1
==B
0
==B
0
SCRT
上升：Q0.5
R
1
定向向上：M30.1
R
1
网络7　无呼梯返回主程序步结束
SCRE
网络8　复位上行返回步，返回主程序
上行返回：S1.1
上行返回：S1.1
R
1
RET
网络9　判断关门步开始
判断关门：S1.2
SCR
网络10　关门
始终闭合：SM0.0
关门限位：I1.4
上升：Q0.5
关门：Q0.4
S
1
网络11　关门到位转换到判断目标步
关门限位：I1.4
关门：Q0.4
R
1
判断目标：S1.3
SCRT
网络12　判断关门步结束
SCRE
网络13　判断目标步开始
判断目标：S1.3
SCR
网络14　下站需要停
楼层指示1：Q0.1
内选信号2：M0.2
下站停步：M2.1
上呼信号1：M0.4
网络15　下站不需要停
楼层指示2：Q0.2
内选信号3：M0.3
下站停步：M2.1
下站不停：M2.2
上呼信号2:M0.5
楼层指示1:Q0.1
内选信号3:M0.3
上呼信号2:M0.5

PLC 控制梯形图（续）

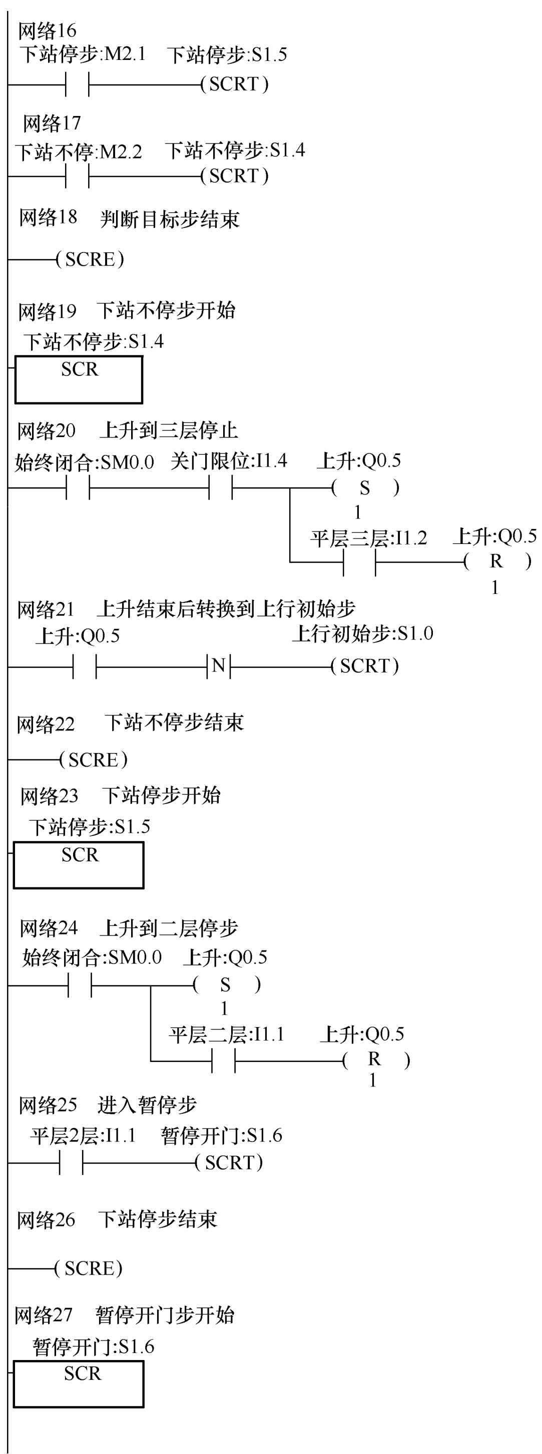
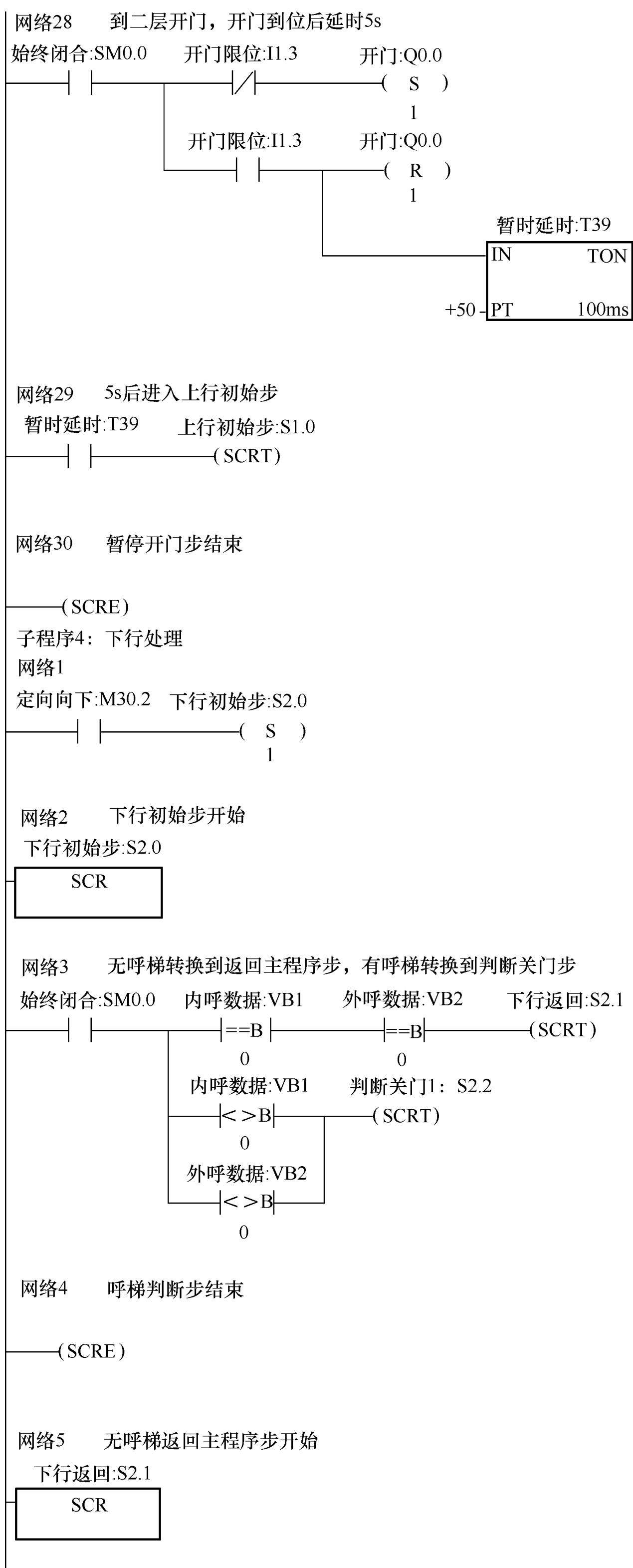

网络16
下站停步:M2.1
下站停步:S1.5
SCRT
网络17
下站不停:M2.2
下站不停步:S1.4
SCRT
网络18 判断目标步结束
SCRE
网络19 下站不停步开始
下站不停步:S1.4
SCR
网络20 上升到三层停止
始终闭合:SM0.0
关门限位:I1.4
上升:Q0.5
S
1
平层三层:I1.2
上升:Q0.5
R
1
网络21 上升结束后转换到上行初始步
上升:Q0.5
N
上行初始步:S1.0
SCRT
网络22 下站不停步结束
SCRE
网络23 下站停步开始
下站停步:S1.5
SCR
网络24 上升到二层停步
始终闭合:SM0.0
上升:Q0.5
S
1
平层二层:I1.1
上升:Q0.5
R
1
网络25 进入暂停步
平层2层:I1.1
暂停开门:S1.6
SCRT
网络26 下站停步结束
SCRE
网络27 暂停开门步开始
暂停开门:S1.6
SCR

图 3.5-25 三层电梯


网络28 到二层开门，开门到位后延时5s
始终闭合:SM0.0
开门限位:I1.3
开门:Q0.0
S
1
开门限位:I1.3
开门:Q0.0
R
1
暂时延时:T39
IN
TON
+50
PT
100ms
网络29 5s后进入上行初始步
暂时延时:T39
上行初始步:S1.0
SCRT
网络30 暂停开门步结束
SCRE
子程序4：下行处理
网络1
定向向下:M30.2
下行初始步:S2.0
S
1
网络2 下行初始步开始
下行初始步:S2.0
SCR
网络3 无呼梯转换到返回主程序步，有呼梯转换到判断关门步
始终闭合:SM0.0
内呼数据:VB1
外呼数据:VB2
下行返回:S2.1
==B
0
==B
0
SCRT
内呼数据:VB1
判断关门1：S2.2
<>B
0
SCRT
外呼数据:VB2
<>B
0
网络4 呼梯判断步结束
SCRE
网络5 无呼梯返回主程序步开始
下行返回:S2.1
SCR

PLC 控制梯形图（续）

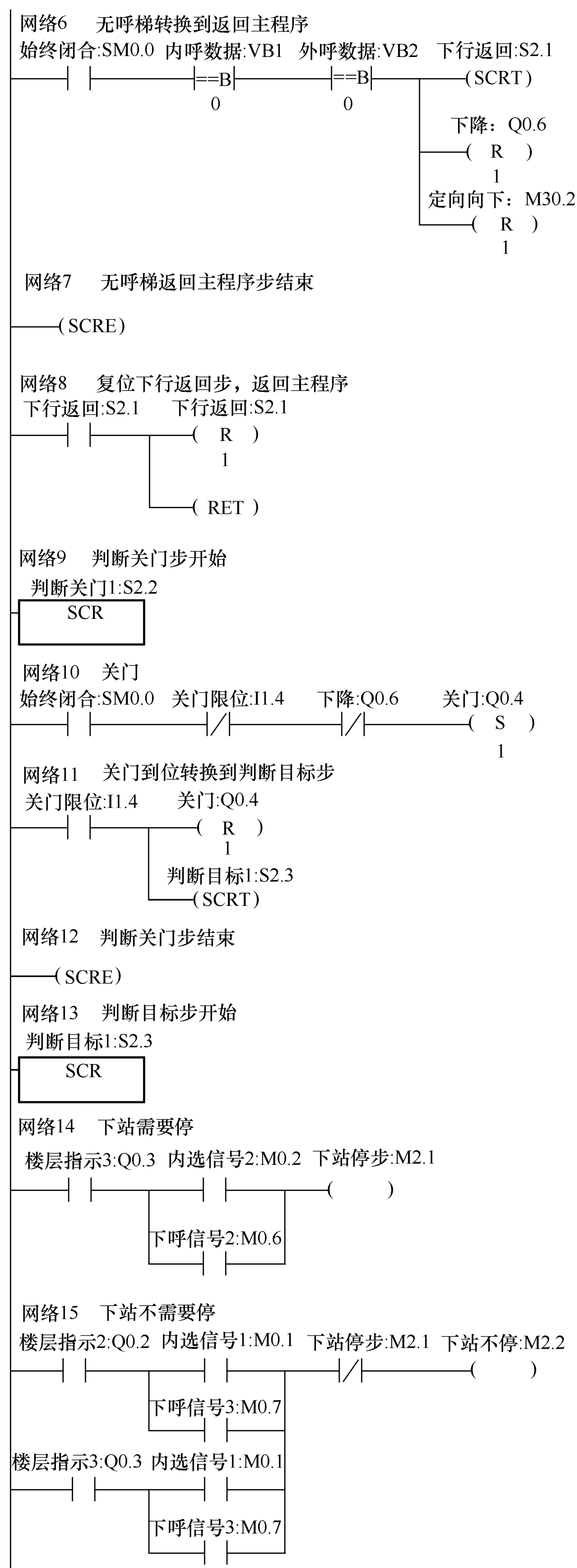
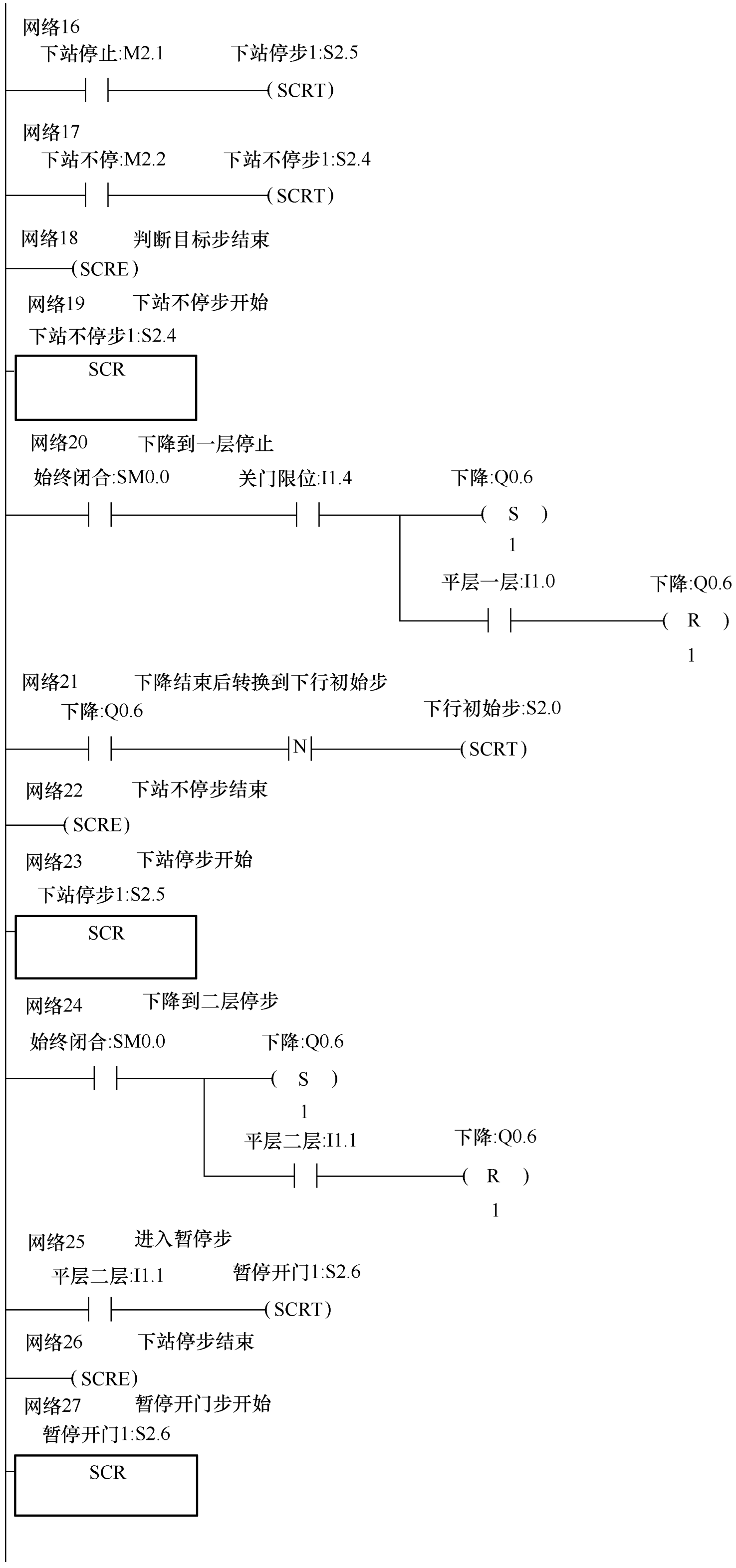

网络6 无呼梯转换到返回主程序
始终闭合:SM0.0
内呼数据:VB1
外呼数据:VB2
下行返回:S2.1
==B
0
==B
0
(SCRT)
下降：Q0.6
(R)
1
定向向下：M30.2
(R)
1
网络7 无呼梯返回主程序步结束
(SCRE)
网络8 复位下行返回步，返回主程序
下行返回:S2.1
下行返回:S2.1
(R)
1
(RET)
网络9 判断关门步开始
判断关门1:S2.2
SCR
网络10 关门
始终闭合:SM0.0
关门限位:I1.4
下降:Q0.6
关门:Q0.4
(S)
1
网络11 关门到位转换到判断目标步
关门限位:I1.4
关门:Q0.4
(R)
1
判断目标1:S2.3
(SCRT)
网络12 判断关门步结束
(SCRE)
网络13 判断目标步开始
判断目标1:S2.3
SCR
网络14 下站需要停
楼层指示3:Q0.3
内选信号2:M0.2
下站停步:M2.1
下呼信号2:M0.6
网络15 下站不需要停
楼层指示2:Q0.2
内选信号1:M0.1
下站停步:M2.1
下站不停:M2.2
下呼信号3:M0.7
楼层指示3:Q0.3
内选信号1:M0.1
下呼信号3:M0.7

图 3. 5-25 三层电梯的


网络16
下站停止:M2.1
下站停步1:S2.5
SCRT
网络17
下站不停:M2.2
下站不停步1:S2.4
SCRT
网络18　判断目标步结束
SCRE
网络19　下站不停步开始
下站不停步1:S2.4
SCR
网络20　下降到一层停止
始终闭合:SM0.0
关门限位:I1.4
下降:Q0.6
S
1
平层一层:I1.0
下降:Q0.6
R
1
网络21　下降结束后转换到下行初始步
下降:Q0.6
N
下行初始步:S2.0
SCRT
网络22　下站不停步结束
SCRE
网络23　下站停步开始
下站停步1:S2.5
SCR
网络24　下降到二层停步
始终闭合:SM0.0
下降:Q0.6
S
1
平层二层:I1.1
下降:Q0.6
R
1
网络25　进入暂停步
平层二层:I1.1
暂停开门1:S2.6
SCRT
网络26　下站停步结束
SCRE
网络27　暂停开门步开始
暂停开门1:S2.6
SCR

PLC 控制梯形图（续）

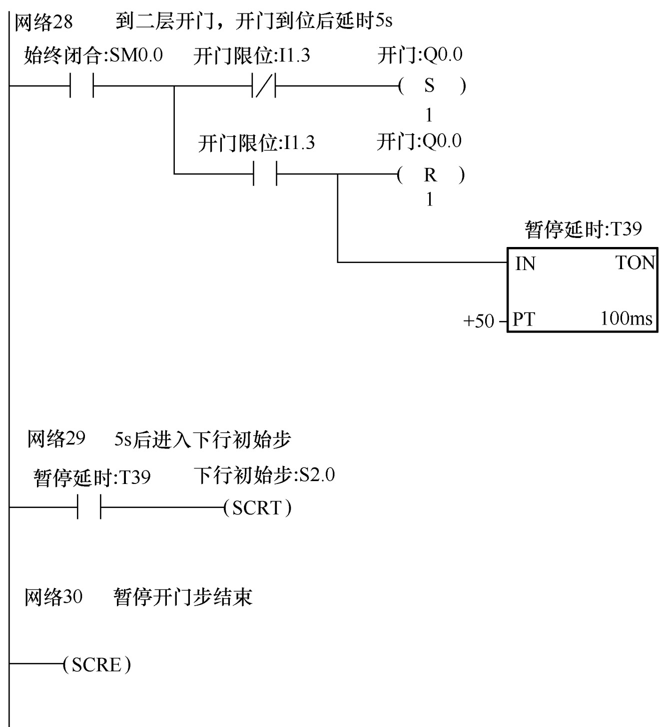


图 3.5-25　三层电梯的 PLC 控制梯形图（续）

第四章

S7-200 PLC 功能指令

第一节　数据传送指令

一、指令介绍

S7-200 CPU 指令系统为存储单元之间数据的传递提供了数据传送指令，数据传送指令有字节、字、双字和实数的单一传送指令，字节立即传送（读和写）指令和以字节、字、双字为单位的数据块的块传送指令。

1. 单一传送指令

单一传送指令可以进行一个数据的传送，数据类型可以是一个字节、字、双字和实数，具体指令介绍见表 4.1-1。

表 4.1-1　单一传送指令表

单一传送指令	梯　形　图	指　　令	指令功能	数据类型及操作数
字节传送	MOV_B EN　ENO ????-IN　OUT-????	MOVB IN，OUT	使能输入（EN）有效时，将 1 个无符号的单字节数据 IN 传送到 OUT 中	数据类型：字节 IN/OUT 操作数：VB、IB、QB、MB、SMB、LB、SB、AC、＊VD、＊AC、＊LD 和常数（注：OUT 操作数没有常数）
字传送	MOV_W EN　ENO ????-IN　OUT-????	MOVW IN，OUT	使能输入（EN）有效时，将 1 个无符号的单字长数据 IN 传送到 OUT 中	数据类型：字 IN/OUT 操作数：VW、IW、QW、MW、SW、SMW、LW、AC、＊VD、＊AC、＊LD、T、C、常数（注：OUT 操作数中没有常数）
双字传送	MOV_DW EN　ENO ????-IN　OUT-????	MOVD IN，OUT	使能输入（EN）有效时，将 1 个有符号的双字长数据 IN 传送到 OUT 中	数据类型：双字 IN/OUT 操作数：VD、ID、QD、MD、SMD、LD、AC、＊VD、＊AC、＊LD、常数（注：OUT 操作数中没有常数）
实数传送	MOV_R EN　ENO ????-IN　OUT-????	MOVR IN，OUT	使能输入（EN）有效时，将 1 个有符号的双字长实数数据 IN 传送到 OUT 中	数据类型：实数 IN/OUT 操作数：VD、ID、QD、MD、SMD、LD、AC、＊VD、＊AC、＊LD、常数（注：OUT 操作数中没有常数）

2. 字节立即传送指令

字节立即传送指令允许在物理 I/O 和存储器之间立即传送一个字节数据。字节立即传送指令包括字节立即读 BIR（Byte Immediately Read）指令和字节立即写 BIW（Byte Immediately Write）指令，其梯形图和语句表见表 4. 1-2。

表 4. 1-2　字节立即传送（读和写）指令的梯形图和语句表

指令名称	梯形图	语句表	指令功能	操作数及数据类型
字节立即读指令	MOV_BIR EN ENO IN OUT	BIR　IN，OUT	BIR 指令读取实际输入端 IN 给出的 1 个字节的数值，并将结果写入 OUT 所指定的存储单元，但输入映像寄存器未更新	IN：IB、*VD、*LD、*AC；数据类型：字节 OUT：VB、IB、QB、MB、SB、SMB、LB、AC、*VD、*AC、*LD；数据类型：字节
字节立即写指令	MOV_BIW EN ENO IN OUT	BIW　IN，OUT	BIW 指令从输入 IN 所指定的存储单元中读取 1 个字节的数值并写入（以字节为单位）实际输出 OUT 端的物理输出点，同时刷新对应的输出映像寄存器	IN：VB、IB、QB、MB、SB、SMB、LB、AC、*VD、*AC、*LD、常数；数据类型：字节 OUT：QB、*VD、*LD、*AC；数据类型：字节

3. 块传送指令

块传送指令用来进行一次传送多个数据，将最多可达 255 个的数据组成 1 个数据块。数据块的类型主要有字节块、字节和双字块，具体指令的介绍见表 4. 1-3。

表 4. 1-3　块传送指令表

块传送指令	梯形图	指令	指令功能	数据类型及操作数
字节块传送	BLKMOV_B EN ENO ????-IN OUT-???? ????-N	BMB IN，OUT，N	使能输入（EN）有效时，把从输入（IN）字节开始的 *N* 个字节数据传送到以输出字节（OUT）开始的 *N* 个字节存储单元中	数据类型：字节 IN/OUT 操作数：VB、IB、QB、MB、SMB、LB、AC、HC、*VD、*AC、*LD N 操作数：VB、IB、QB、MB、SMB、LB、AC、*VD、*AC、*LD
字块传送	BLKMOV_W EN ENO ????-IN OUT-???? ????-N	BMW IN，OUT，N	使能输入（EN）有效时，把从输入（IN）字节开始的 *N* 个字节数据传送到以输出字节（OUT）开始的 *N* 个字存储单元中	数据类型：字 IN/OUT 操作数：VW、IW、QW、MW、AIW、SMW、LW、AC、AQW、HC、*VD、*AC、*LD、T、C N 数据类型：字节；操作数：VB、IB、QB、MB、SMB、LB、AC、*VD、*AC、*LD
双字块传送	BLKMOV_D EN ENO ????-IN OUT-???? ????-N	BMD IN，OUT，N	使能输入（EN）有效时，把从输入（IN）字节开始的 *N* 个字节数据传送到以输出字节（OUT）开始的 *N* 个双字存储单元中	数据类型：双字 IN/OUT 操作数：VD、ID、QD、MD、SMD、LD、AC、*VD、*AC、*LD N 数据类型：字节；操作数：VB、IB、QB、MB、SMB、LB、AC、*VD、*AC、*LD 和常数

二、指令练习

1. 字节传送指令练习

编写一段程序，将常数 88 传送到 VB0 中。程序如图 4. 1-1 所示，字节 VB0 中的数据为 88。注意：若将输出 VB0 改成 VW0，则程序出错，因为单字节传送的操作数不能为字。

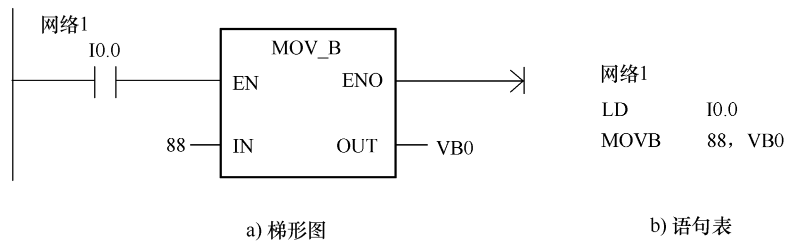


图 4. 1-1　单字节传送程序示例

2. 字传送指令练习

将十六进制数 16#E071 传送到 QW0 中，程序如图 4. 1-2 所示，则字节 QB0 中的数据为 2#11100000，字节 QB1 中的数据为 2#01110001。若将输出 QW0 改成 QB0，则程序出错，因为单字传送的操作数不能为字节。

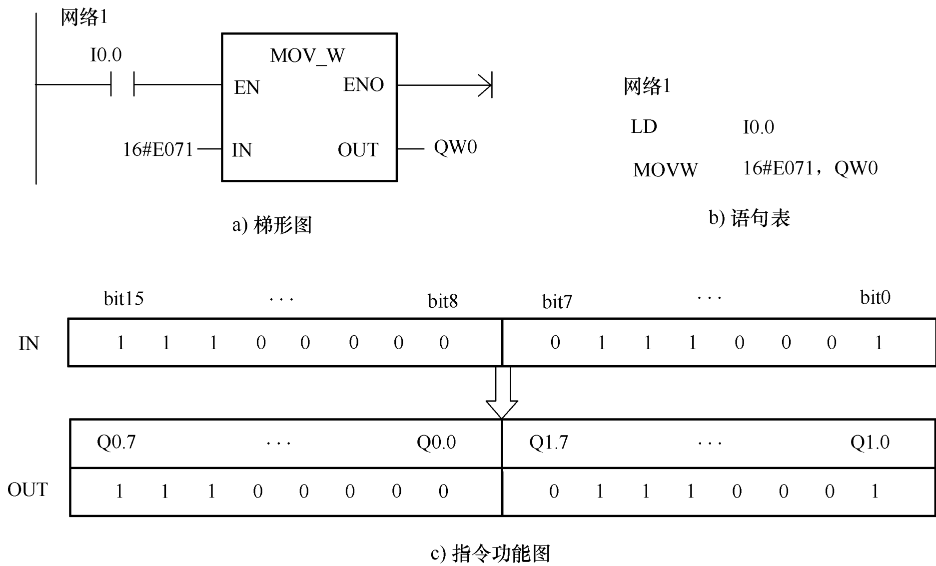


图 4. 1-2　单字传送程序示例

3. 字节块传送指令练习

编写一段程序，将以 VB0 开始的 4 个字节的内容传送至以 VB10 开始的 4 个字节存储单元中，VB0 ~ VB3 的数据分别为 2、3、4、5。程序如图 4. 1-3 所示。当 I0. 0 接通后，程序运行结果见表 4. 1-4。

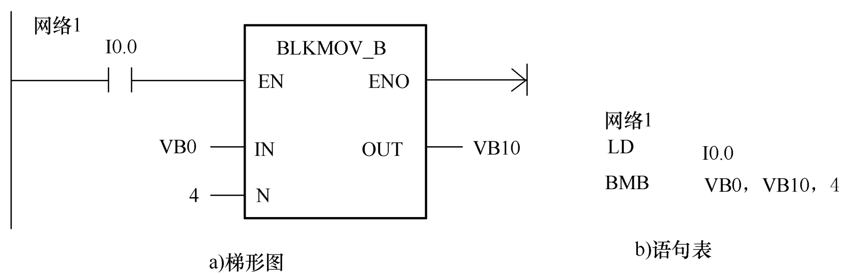


图 4.1-3　字节块传送程序示例

表 4.1-4　运行结果示意表

数组 1 的数据	2	3	4	5
数据地址	VB0	VB1	VB2	VB3
数组 2 的数据	2	3	4	5
数据地址	VB10	VB11	VB12	VB13

4. 传送指令综合练习

存储器初始化程序是用于 PLC 开机运行时对某些存储器清 0 或设置的一种操作，常采用传送指令来编程。若开机运行时将 VB0 清 0，将 VW20 设置为 200，则对应的梯形图程序如图 4.1-4 所示。

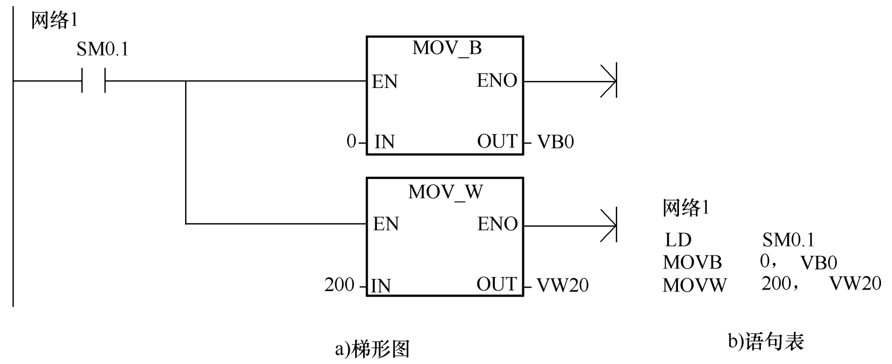


图 4.1-4　存储器的清 0 与设置

第二节　比较操作指令

一、指令介绍

比较指令是在指定的条件下比较两个操作数的大小，条件成立时，触点闭合。在实际应用中，比较指令为上、下限位控制以及数值条件判断提供了方便。比较指令包括相等指令（= =）、小于等于指令（< =）、大于等于指令（> =）、大于（>）、小于（<）、不等指令（< >）等。比较指令的类型有字节比较、整数比较、双字整数比较、实数比较和字符串

比较，下面主要介绍数值比较指令（见表4.2-1）。

表4.2-1　数值比较指令表

指令名称	梯形图	语句表	指令功能	操作数范围及数据类型
字节比较指令	IN1 ┤ >B ├ IN2	LDB > IN1，IN2 AB > IN1，IN2 OB > IN1，IN2	字节比较用于比较两个字节型整数值 IN1 和 IN2 的大小，条件满足时，触点接通。字节比较是无符号的	IN1、IN2：IB、QB、MB、SMB、VB、SB、LB、AC、＊VD、＊AC、＊LD、常数；数据类型：字节 OUT：I、Q、M、SM、T、C、V、S、L、使能位；数据类型：位
	IN1 ┤ >=B ├ IN2	LDB > = IN1，IN2 AB > = IN1，IN2 OB > = IN1，IN2		
	IN1 ┤ ==B ├ IN2	LDB = IN1，IN2 AB = IN1，IN2 OB = IN1，IN2		
	IN1 ┤ <=B ├ IN2	LDB < = IN1，IN2 AB < = IN1，IN2 OB < = IN1，IN2		
	IN1 ┤ <B ├ IN2	LDB < IN1，IN2 AB < IN1，IN2 OB < IN1，IN2		
	IN1 ┤ <>B ├ IN2	LDB < > IN1，IN2 AB < > IN1，IN2 OB < > IN1，IN2		
整数比较指令	IN1 ┤ >I ├ IN2	LDW > IN1，IN2 AW > IN1，IN2 OW > IN1，IN2	整数比较用于比较两个一个字长的整数值 IN1 和 IN2 的大小，条件满足时，触点接通。整数比较是有符号的（最高位为符号位），其范围是16#8000～16#7FFF。例如，16#7FFF > 16#8000（后者为负数）	IN1、IN2：IW、QW、MW、SW、SMW、T、C、VW、LW、AIW、AC、＊VD、＊LD、＊AC、常数；数据类型：整数 OUT：I、Q、M、SM、T、C、V、S、L、使能位；数据类型：位
	IN1 ┤ >=I ├ IN2	LDW > = IN1，IN2 AW > = IN1，IN2 OW > = IN1，IN2		
	IN1 ┤ ==I ├ IN2	LDW = IN1，IN2 AW = IN1，IN2 OW = IN1，IN2		
	IN1 ┤ <=I ├ IN2	LDW < = IN1，IN2 AW < = IN1，IN2 OW < = IN1，IN2		
	IN1 ┤ <I ├ IN2	LDW < IN1，IN2 AW < IN1，IN2 OW < IN1，IN2		
	IN1 ┤ <>I ├ IN2	LDW < > IN1，IN2 AW < > IN1，IN2 OW < > IN1，IN2		

（续）

指令名称	梯形图	语句表	指令功能	操作数范围及数据类型
双字整数比较指令	IN1 ┤ >D ├ IN2	LDD >IN1, IN2 AD >IN1, IN2 OD >IN1, IN2	双字整数比较用于比较两个双字长整数值 IN1 和 IN2 的大小，条件满足时，触点接通。它们的比较也是有符号的（最高位为符号位），其范围是 16#80000000 ~ 16#7FFFFFFF。例如，16#7FFFFFFF >16#80000000（后者为负数）	IN1、IN2：ID、QD、MD、SD、SMD、VD、LD、HC、AC、*VD、*LD、*AC、常数；数据类型：双整数 OUT：I、Q、M、SM、T、C、V、S、L、使能位；数据类型：位
	IN1 ┤ >=D ├ IN2	LDD > =IN1, IN2 AD > =IN1, IN2 OD > =IN1, IN2		
	IN1 ┤ ==D ├ IN2	LDD =IN1, IN2 AD =IN1, IN2 OD =IN1, IN2		
	IN1 ┤ <=D ├ IN2	LDD < =IN1, IN2 AD < =IN1, IN2 OD < =IN1, IN2		
	IN1 ┤ <D ├ IN2	LDD <IN1, IN2 AD <IN1, IN2 OD <IN1, IN2		
	IN1 ┤ <>D ├ IN2	LDD < >IN1, IN2 AD < >IN1, IN2 OD < >IN1, IN2		
实数比较指令	IN1 ┤ >R ├ IN2	LDR >IN1, IN2 AR >IN1, IN2 OR >IN1, IN2	实数比较用于比较两个双字长实数值 IN1 和 IN2 的大小，条件满足时，触点接通。实数比较是有符号的（最高位为符号位），负实数范围是 -1.175495E-38 ~ -3.402823E+38，正实数范围是 +1.175495E-38 ~ +3.402823E+38	IN1、IN2：ID、QD、MD、SD、SMD、VD、LD、AC、*VD、*LD、*AC、常数；数据类型：实数 OUT：I、Q、M、SM、T、C、V、S、L、使能位；数据类型：位
	IN1 ┤ >=R ├ IN2	LDR > =IN1, IN2 AR > =IN1, IN2 OR > =IN1, IN2		
	IN1 ┤ ==R ├ IN2	LDR =IN1, IN2 AR =IN1, IN2 OR =IN1, IN2		
	IN1 ┤ <=R ├ IN2	LDR < =IN1, IN2 AR < =IN1, IN2 OR < =IN1, IN2		
	IN1 ┤ <R ├ IN2	LDR <IN1, IN2 AR <IN1, IN2 OR <IN1, IN2		
	IN1 ┤ <>R ├ IN2	LDR < >IN1, IN2 AR < >IN1, IN2 OR < >IN1, IN2		

二、指令练习

1. 字节比较指令练习

如图 4.2-1 所示，用接通延时定时器和数值比较指令可组成占空比可调的脉冲发生器（断电 6s、通电 4s）；Q0.0 为 0 的时间取决于数值比较指令（LDW >= T37，60）中的第 2 个操作数的值。

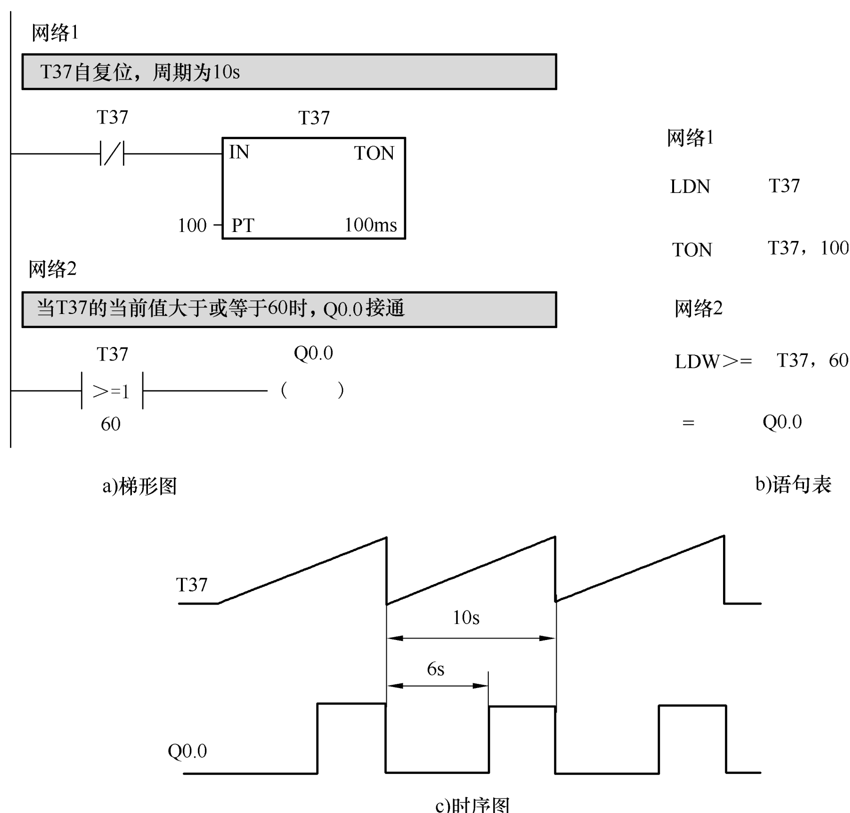


图 4.2-1　接通延时定时器和数值比较指令组成脉冲发生器

2. 整数比较指令练习

练习 1：图 4.2-2 所示为变量存储器 VW100 中的数值与十进制数 50 相比较。当变量存储器 VW100 中的数值等于十进制数 50 时，常开触点接通，线圈 Q0.0 有输出。

练习 2：图 4.2-3 所示为整数比较指令应用的梯形图程序。当 I0.0 触点由断开到接通时，CU 端接收一次脉冲，计数器的值加 1；当计数值等于 2 时，即整数比较指令 C5 = 2 时，触点接通，Q0.0 接通；当复位（R）端的 I0.1 接通时，C5 计数器复位，当前值清零。

练习 3：图 4.2-4 所示为用比较指令来实现电动机的起动与停止的梯形图程序。按下起动按钮 SB1，第一台电动机起动运转，5s 以后第二台电动机起动运转，再过 5s 后第三台电动机起动运转；按下停止按钮 SB2，第三台电动机停止，随后过 5s 第二台电动机停止，再过 5s 第一台电动机停止。

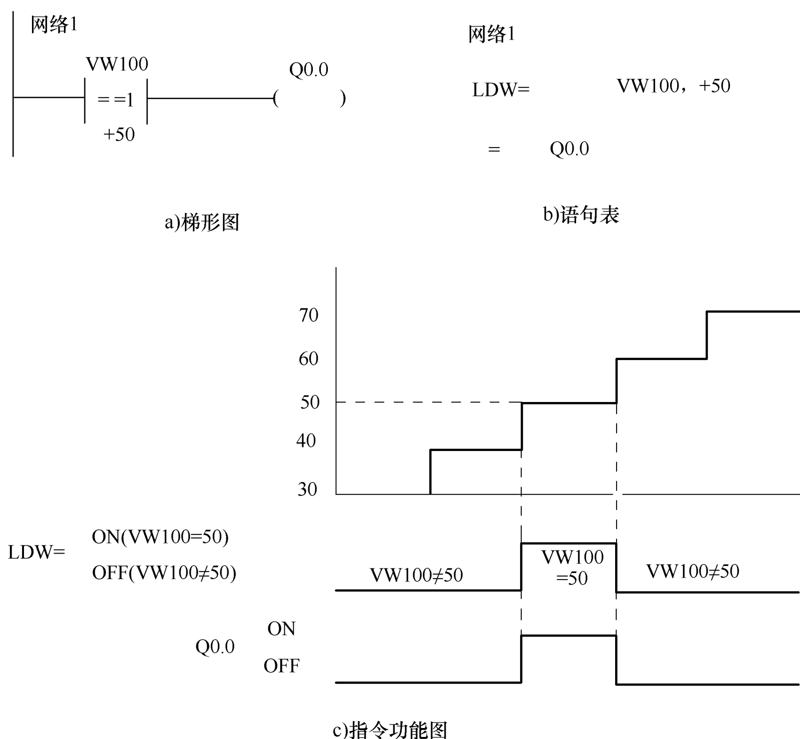


图 4.2-2　整数比较指令的应用程序

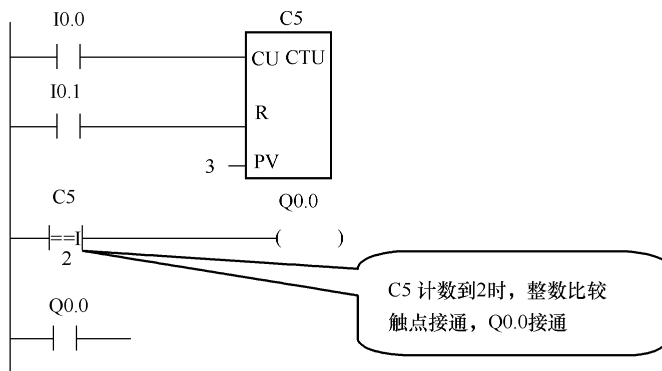


图 4.2-3　整数比较指令的应用

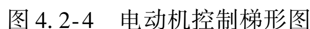


图 4.2-4　电动机控制梯形图

第三节　循 环 指 令

一、指令介绍

循环指令的引入为解决重复执行相同功能的程序提供了极大方便，并且优化了程序结构，特别在进行大量相同功能的计算和逻辑处理时，循环指令非常有用。循环指令有两条：循环开始指令（FOR）和循环结束指令（NEXT），具体指令介绍见表 4.3-1。

表 4.3-1　循环指令表

指令类型	梯形图	指　令	指令功能	数据类型及操作数
循环开始 FOR	FOR EN ENO ????-INDX ????-INIT ????-FINAL	FOR INDX, INIT, FINAL	FOR 指令为指令盒格式，主要执行 FOR 和 NEXT 之间的指令，使能输入 EN，当前值计数器 INDX，循环次数初始值 INIT，循环计数终值 FINAL	数据类型：整数 INDX：VW、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC FINAL 和 INIT：VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常量、*VD、*AC、*LD
循环结束 NEXT	——(NEXT)	NEXT	循环结束	无

二、指令练习

图 4.3-1 所示为循环指令应用的梯形图及语句表。当 I0.0 接通时，外层循环执行 100 次；当 I0.1 接通时，内层循环执行 2 次。

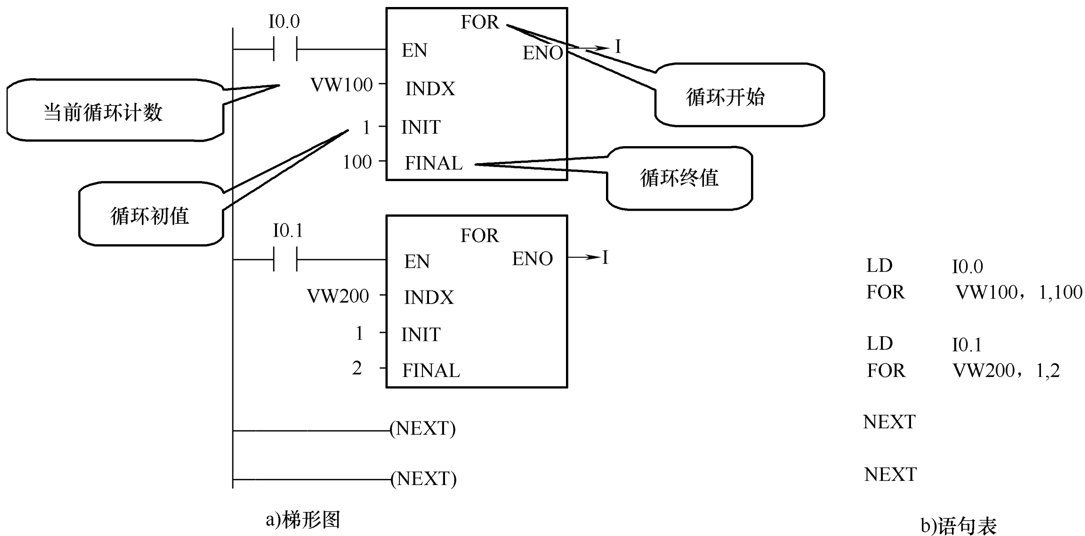


图 4.3-1　循环指令的应用

使用说明：

1）FOR 指令必须与 NEXT 指令配套使用。

2）指令允许循环嵌套使用，最多可嵌套 8 层。

3）每次启用循环指令时，将初始值 INIT 传送到循环计数 INDX 中。

第四节　移 位 指 令

一、指令介绍

移位指令在 PLC 控制中是比较常用的，根据移位的数据长度可分为字节型移位、字型移位和双字型移位；根据移位的方向可分为左移和右移，还可进行循环左移和循环右移。

1. 左/右移位指令

移位指令的功能是将 IN 中的数的各位向左或向右移动 N 位后，送给 OUT。移位指令对移出的位自动补 0。如果移位的位数 N 大于或等于允许值（如字节操作为 8，字操作为 16，双字操作为 32），应对 N 进行取模操作。例如，对于字移位，将 N 除以 16 后取余数，从而得到一个有效的移位次数。取模操作的结果对于字节操作是 0 ~ 7，对于字操作是 0 ~ 15，对于双字操作是 0 ~ 31。如果 N 分别等于 8、16、32，则不进行移位操作。所有的移位指令中的 N 均为字节型数据。如果移位次数大于 0，“溢出”存储器位 SM1.1 保存最后一次被移出的位的值。如果移出结果为 0，零标志位 SM1.0 被置 1。移位指令的具体介绍见表 4.4-1。

表 4.4-1　移位指令表

移位指令	梯形图	指　令	指令功能	数据类型及操作数
字节左移 SLB	SHL_B EN ENO ????-IN OUT-???? ????-N	SLB OUT，N	当输入 EN 有效时，将字节型输入数据 IN 左移 N 位（N≤8）后，右端移出位补 0，送到 OUT 指定的字节存储单元	数据类型：字节 IN/N：VB、IB、QB、MB、SMB、LB、AC、*VD、*AC、*LD、常数 OUT：VB、IB、QB、MB、SMB、LB、AC、*VD、*AC、*LD
字节右移指令 SRB	SHR_B EN ENO ????-IN OUT-???? ????-N	SRB OUT，N	当输入 EN 有效时，将字节型输入数据 IN 右移 N 位（N≤8）后，左端移出位补 0，送到 OUT 指定的字节存储单元	
字左移指令 SLW	SHL_W EN ENO ????-IN OUT-???? ????-N	SLW OUT，N	当允许输入 EN 有效时，将字型输入数据 IN 左移 N 位（N≤16）后，送到 OUT 指定的字存储单元	IN/OUT 数据类型：字 操作数：VW、IW、QW、MW、SW、SMW、LW、AIW、AC、*VD、*AC、*LD、常数（注：OUT 操作数中没有常数） N 数据类型：字节 操作数：VB、IB、QB、MB、SB、SMB、LB、AC、*VD、*AC、*LD、常数
字右移指令 SRW	SHR_W EN ENO ????-IN OUT-???? ????-N	SRW OUT，N	当输入 EN 有效时，将字型输入数据 IN 右移 N 位（N≤16）后，送到 OUT 指定的字存储单元	
双字左移指令 SLD	SHL_DW EN ENO ????-IN OUT-???? ????-N	SLD OUT，N	当输入 EN 有效时，将双字型输入数据 IN 左移 N 位（N≤32）后，送到 OUT 指定的双字存储单元	IN/OUT 数据类型：双字 操作数：VD，ID，QD，MD，SD，SMD，LD，HC，AC，*VD，*AC，*LD 和常数（注：OUT 操作数中没有常数） N 数据类型：字节 操作数：VB、IB、QB、MB、SB、SMB、LB、AC、*VD、*AC、*LD、常数
双字右移指令 SRD	SHR_DW EN ENO ????-IN OUT-???? ????-N	SRD OUT，N	当输入 EN 有效时，将双字型输入数据 IN 右移 N 位（N≤32）后，送到 OUT 指定的双字存储单元	

2. 循环左/右移位指令

循环移位指令的功能是将 IN 中的数的各位向左或向右循环移动 N 位后，送给 OUT。循环移位是环形的，即被移出来的位将返回到另一端空出来的位置，具体指令介绍见表 4.4-2。如果移动的位数 N 大于或等于允许值（字节操作为 8，字操作为 16，双字操作为 32），执行循环移位之前先对 N 进行取模操作。如果取模操作的结果为 0，不进行循环移位操作。

如果循环移位指令被执行，移出的最后一位的数值会被复制到溢出标志位（SM1.1）。

如果实际移位次数为 0，零标志位（SM1.0）被置为 1。

另外，字节操作是无符号的，对于字和双字操作，当使用有符号数据类型时，符号位也被移位。

表 4.4-2　循环移位指令表

循环移位指令	梯　形　图	指　　令	指 令 功 能	数据类型及操作数
字节循环左移指令 ROL-B	ROL_B EN　ENO ????-IN　OUT-???? ????-N	SLB OUT，N	当输入 EN 有效时，将字节型输入数据 IN 左移 N 位（N≤8）后，右端移出位补 0，送到 OUT 指定的字节存储单元	数据类型：字节 IN/N：VB、IB、QB、MB、SMB、LB、AC、*VD、*AC、*LD、常数 OUT：VB、IB、QB、MB、SMB、LB、AC、*VD、*AC、*LD
字节循环右移指令 ROR-B	ROR_B EN　ENO ????-IN　OUT-???? ????-N	SRB OUT，N	当输入 EN 有效时，将字节型输入数据 IN 右移 N 位（N≤8）后，左端移出位补 0，送到 OUT 指定的字节存储单元	
字循环左移指令 ROL-W	ROL_W EN　ENO ????-IN　OUT-???? ????-N	SLW OUT，N	当允许输入 EN 有效时，将字型输入数据 IN 左移 N 位（N≤16）后，送到 OUT 指定的字存储单元	IN/OUT 数据类型：字 操作数：VW、IW、QW、MW、SW、SMW、LW、AIW、AC、*VD、*AC、*LD、常数（注：OUT 操作数中没有常数） N 数据类型：字节 操作数：VB、IB、QB、MB、SB、SMB、LB、AC、*VD、*AC、*LD、常数
字循环右移指令 ROR-W	ROR_W EN　ENO ????-IN　OUT-???? ????-N	SRW OUT，N	当输入 EN 有效时，将字型输入数据 IN 右移 N 位（N≤16）后，送到 OUT 指定的字存储单元	
双字循环左移指令 ROL-DW	ROL_DW EN　ENO ????-IN　OUT-???? ????-N	SLD OUT，N	当输入 EN 有效时，将双字型输入数据 IN 左移 N 位（N≤32）后，送到 OUT 指定的双字存储单元	IN/OUT 数据类型：双字 操作数：VD，ID，QD，MD，SD，SMD，LD，HC，AC，*VD，*AC，*LD和常数（注：OUT 操作数中没有常数） N 数据类型：字节 操作数：VB、IB、QB、MB、SB、SMB、LB、AC、*VD、*AC、*LD、常数
双字循环右移指令 ROR-DW	ROR_DW EN　ENO ????-IN　OUT-???? ????-N	SRD OUT，N	当输入 EN 有效时，将双字型输入数据 IN 右移 N 位（N≤32）后，送到 OUT 指定的双字存储单元	

3. 寄存器移位指令

寄存器移位 SHRB（SHift Register Bit）指令是一个移位长度可以指定的移位指令，EN 使能端连接移位脉冲信号，每次使能有效时，整个移位寄存器移动 1 位。DATA 为数据输入端，连接移入移位寄存器的二进制数值，执行指令时将该位的值移入寄存器。S_BIT 指定移

位寄存器的最低位。N 指定移位寄存器的长度和移位方向，移位寄存器的最大长度为 64 位，N 为正值表示左移位，输入数据（DATA）移入移位寄存器的最低位（S_BIT），并移出移位寄存器的最高位。SHRB 指令介绍见表 4. 4-3。

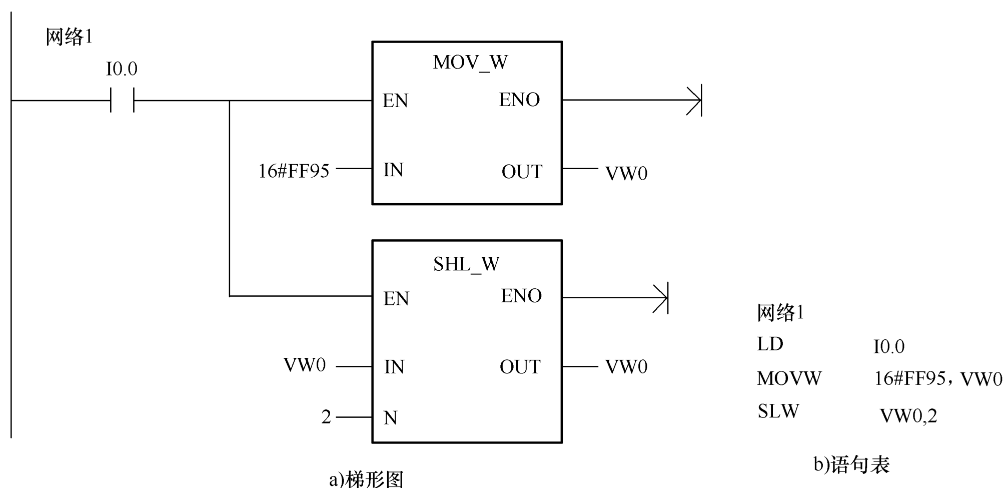
表 4. 4-3 SHRB 指令的梯形图和语句表

指令名称	梯形图	语句表	指令功能	操作数
SHRB 指令	SHRB EN ENO DATA S_BIT N	SHRB DATA，S_BIT，N	EN 使能输入端有效，SHRB 指令将 DATA 数值移入移位寄存器	DATA、S_BIT：I、Q、M、SM、T、C、V、S、L；数据类型：布尔 N：VB、IB、QB、MB、SB、SMB、LB、AC；常数、*VD、*LD、*AC；数据类型：字节

二、指令练习

1. 字左移指令练习

图 4. 4-1 所示为一个移位指令的应用练习。



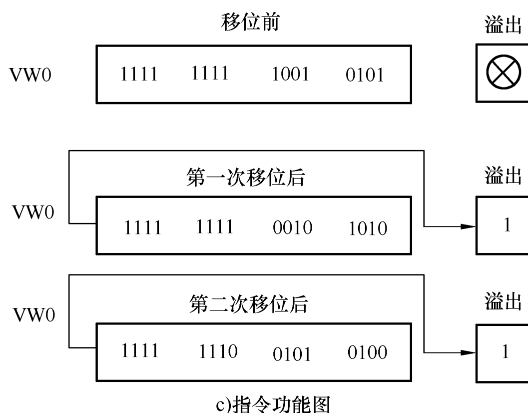


图 4. 4-1 左移位指令的应用示例

移位使用说明：

1）被移位的数据是无符号的。

2）在移位时，存放被移位数据的编程元件的移出端与特殊继电器 SM1.1 连接，移出位进入 SM1.1（溢出），另一端自动补 0。

3）移位次数 N 与移位数据的长度有关，如果 N 小于实际的数据长度，则执行 N 次移位；如果 N 大于或等于允许值（字节操作为 8，字操作为 16，双字操作为 32），则执行移位前先对 N 进行取模操作。如果取模操作的结果为 0，不进行循环移位操作。

2. 字节移位指令练习

如图 4.4-2 所示，当 I0.0 输入有效时，将 VB10 左移 4 位送到 VB10，将 VB0 循环右移 3 位送到 VB0。

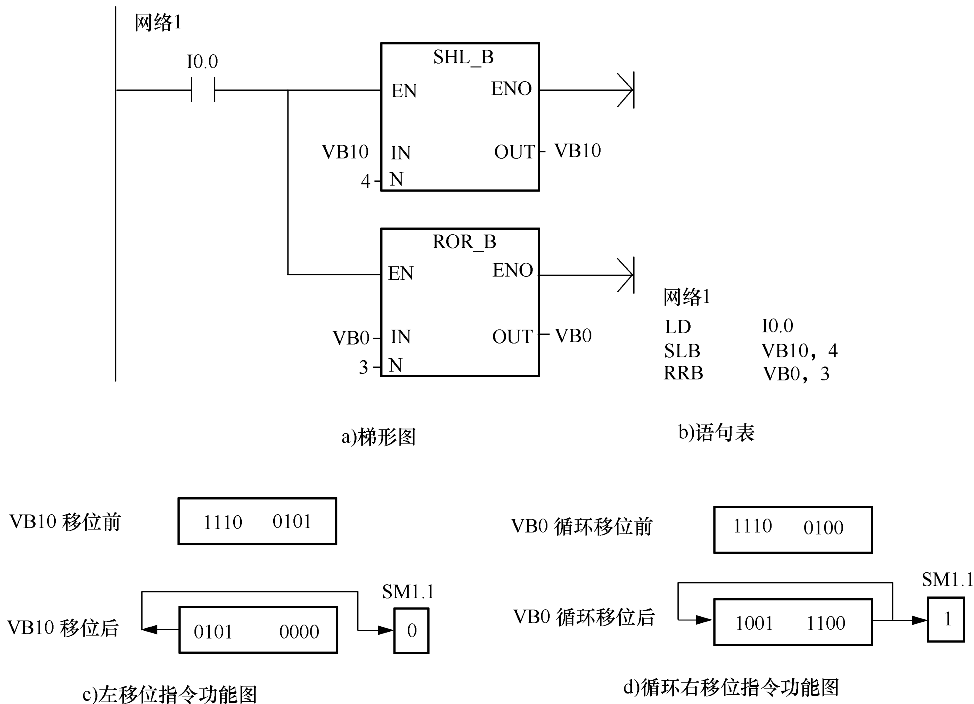


图 4.4-2　字节移位指令程序示例

3. 字循环右移指令练习

如图 4.4-3 所示，将 AC0 中的字循环右移 2 位，将 VW200 中的字左移 3 位。当 I1.0 接通时，将 AC0 中的数据 0100 0000 0000 0001 向右循环移动 2 位变为 0101 0000 0000 0000，同时将 VW200 中的数据 1110 0010 1010 1101 向左移动 3 位变为 0001 0101 0110 1000。

循环移位使用说明：

1）字节操作是无符号的，对于字和双字操作，当使用有符号数据类型时，符号位也被移位。

2）在移位时，存放被移位数据的编程元件的移出端既与另一端连接，又与特殊继电器 SM1.1 连接，移出位在被移到另一端的同时，也进入 SM1.1（溢出）。

3）移位次数 N 与移位数据的长度有关，如果 N 小于实际的数据长度，则执行 N 次移位；如果 N 大于数据长度，则执行移位的次数等于实际数据长度的位数。

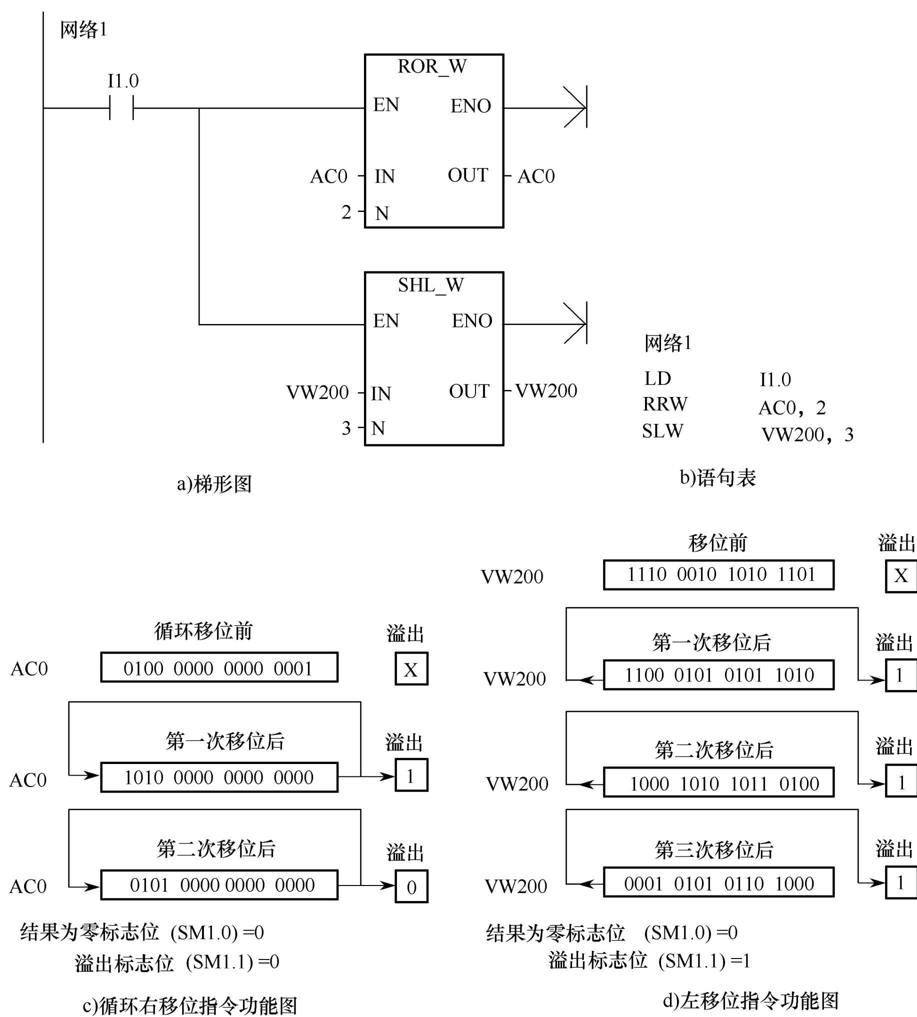


图 4.4-3　字移位指令程序示例

4. 寄存器移位指令练习

图 4.4-4 所示为寄存器移位指令简单的应用示例，每当 I0.1 接通时，产生一个正向脉冲，从而引发一次移位，低位读入 I0.3 的状态数值（高或低），高位则溢出到 SM1.1 特殊寄存器。

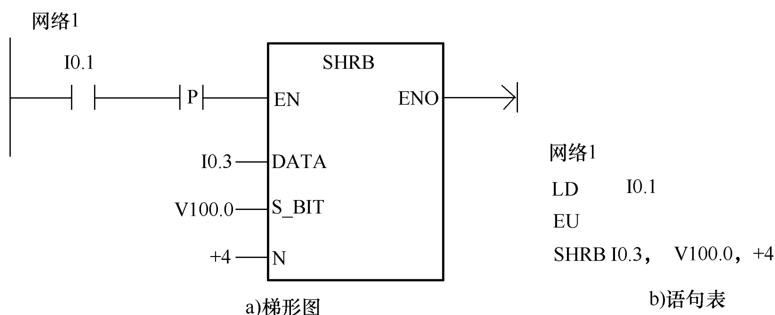


图 4.4-4　SHRB 指令的应用

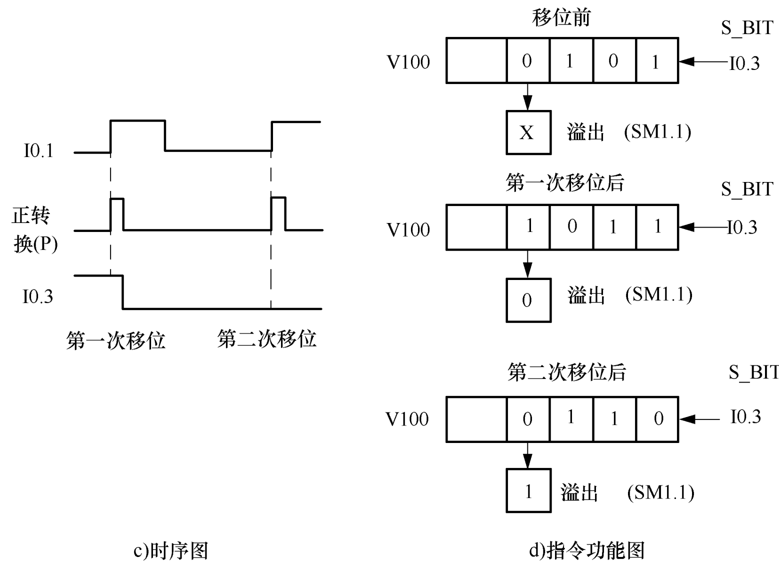


图 4.4-4　SHRB 指令的应用（续）

第五节　四则运算指令

一、指令介绍

1. 增减指令

增减指令又称为自动加 1 或自动减 1 指令，数据的长度有字节、字、双字，指令的具体介绍见表 4.5-1。

表 4.5-1　增减指令表

增减指令	梯形图	指令	功能	数据类型及操作数
字节加 1	INC_B: EN ENO; ????-IN OUT-????	INCB OUT	使能输入有效时，把一字节长的无符号输入数（IN）加 1，得到一字节的无符号输出结果 OUT。执行的结果：IN + 1 = OUT	数据类型：字节 IN 操作数：VB、MB、IB、QB、SB、SMB、LB、AC、＊VD、＊AC、＊LD、常数 OUT：VB、IB、QB、MB、SMB、SB、LB、AC、＊VD、＊AC、＊LD
字节减 1	DEC_B: EN ENO; ????-IN OUT-????	DECB OUT	使能输入有效时，把一字节长的无符号输入数（IN）减 1，得到一字节的无符号输出结果 OUT。执行的结果：IN − 1 = OUT	
字加 1	INC_W: EN ENO; ????-IN OUT-????	INCW OUT	使能输入有效时，把一字长的有符号输入数（IN）加 1，得到一字长的有符号输出结果 OUT。执行的结果：IN + 1 = OUT	数据类型：字 IN 操作数：VW、MW、IW、QW、SW、SMW、LW、AC、＊VD、＊AC、＊LD、常数 OUT：VW、IW、QW、MW、SW、SMW、LW、AC、＊VD、＊AC、＊LD、
字减 1	DEC_W: EN ENO; ????-IN OUT-????	DECW OUT	使能输入有效时，把一字长的有符号输入数（IN）减 1，得到一字长的有符号输出结果 OUT。执行的结果：IN − 1 = OUT	

（续）

增减指令	梯形图	指令	功能	数据类型及操作数
双字加 1	INC_DW: EN ENO; ????-IN OUT-????	INCD OUT	使能输入有效时，把双字长（32 位）的有符号输入数（IN）加 1，得到双字长的有符号输出结果 OUT。IN + 1 = OUT	数据类型：双字 IN 操作数：VD、MD、ID、QD、SD、SMD、LD、AC、*VD、*AC、*LD、常数 OUT：VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD
双字减 1	DEC_DW: EN ENO; ????-IN OUT-????	DECD OUT	使能输入有效时，把双字长的有符号输入数（IN）减 1，得到双字长的有符号输出结果 OUT。IN − 1 = OUT	

2. 加法/减法指令

加法/减法指令是对两个有符号数进行加减操作，可分为整数加减、双整数加减和实数加减指令，具体指令的介绍见表 4.5-2。

表 4.5-2 加法/减法指令表

加法/减法指令	梯形图	指令	功能	数据类型及操作数
整数加法	ADD_I: EN ENO; ????-IN1 OUT-????; ????-IN2	+I IN1，OUT	使能输入有效时，将两个单字长（16 位）的符号整数 IN1 和 IN2 相加，产生一个 16 位整数结果 OUT	数据类型：整数 IN1/IN2 操作数：VW、MW、IW、QW、SW、SMW、LW、AC、*VD、*AC、*LD、T、C、AIW、常数 OUT：VW、IW、QW、MW、SW、SMW、LW、AC、T、C、*VD、*AC、*LD
整数减法	SUB_I: EN ENO; ????-IN1 OUT-????; ????-IN2	−I IN2，OUT	使能输入有效时，将两个单字长的符号整数 IN1 和 IN2 相减，产生一个 16 位整数结果 OUT	
双整数加法	ADD_DI: EN ENO; ????-IN1 OUT-????; ????-IN2	+D IN1，OUT	使能输入有效时，将两个双字长（32 位）的符号整数 IN1 和 IN2 相加，产生一个 32 位双整数结果 OUT	数据类型：双整数 IN1/IN2 操作数：VD、MD、ID、QD、SD、SMD、LD、AC、*VD、*AC、*LD、HC、常数 OUT：VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD
双整数减法	SUB_DI: EN ENO; ????-IN1 OUT-????; ????-IN2	−D IN2，OUT	使能输入有效时，将两个双字长（32 位）的符号整数 IN1 和 IN2 相减，产生一个 32 位双整数结果 OUT	
实数加法	ADD_R: EN ENO; ????-IN1 OUT-????; ????-IN2	+R IN1，OUT	使能输入有效时，将两个双字长（32 位）的实数 IN1 和 IN2 相加，产生一个 32 位实数结果 OUT	数据类型：实数 IN1/IN2 操作数：VD、MD、ID、QD、SD、SMD、LD、AC、*VD、*AC、*LD、常数 OUT：VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD
实数减法	SUB_R: EN ENO; ????-IN1 OUT-????; ????-IN2	−R IN2，OUT	使能输入有效时，将两个双字长（32 位）的实数 IN1 和 IN2 相减，产生一个 32 位实数结果 OUT	

3. 乘/除法指令

乘/除法指令是对两个有符号数进行相乘/除运算，可分为整数乘/除法指令、双整数乘/除法指令、完全整数乘/除法指令及实数乘/除法指令，具体指令的介绍见表 4.5-3。

表 4.5-3　乘/除法指令表

乘/除法指令	梯　形　图	指　　令	功　　能	数据类型及操作数
整数乘法	MUL_I EN ENO ????-IN1 OUT-???? ????-IN2	×I IN1，OUT	使能输入有效时，将两个单字长（16 位）的符号整数 IN1 和 IN2 相乘，产生一个 16 位整数结果 OUT，这里 IN2 与 OUT 是同一个存储单元	数据类型：整数 IN1/IN2 操作数：VW、MW、IW、QW、SW、SMW、LW、AC、*VD、*AC、*LD、T、C、AIW、常数 OUT：VW、IW、QW、MW、SW、SMW、LW、AC、T、C、*VD、*AC、*LD
整数除法	DIV_I EN ENO ????-IN1 OUT-???? ????-IN2	/I IN2，OUT	使能输入有效时，用字型有符号整数 IN1 除以 IN2，产生一个字整数商 OUT，这里 IN1 与 OUT 是同一个存储单元	
双整数乘法	MUL_DI EN ENO ????-IN1 OUT-???? ????-IN2	×D IN1，OUT	使能输入有效时，将两个单字长（16 位）的符号整数 IN1 和 IN2 相乘，产生一个双字整数结果 OUT，这里 IN2 与 OUT 是同一个存储单元	数据类型：双整数 IN1/IN2 操作数：VD、MD、ID、QD、SD、SMD、LD、AC、*VD、*AC、*LD、HC、常数 OUT：VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD
双整数除法	DIV_DI EN ENO ????-IN1 OUT-???? ????-IN2	/D IN2，OUT	使能输入有效时，将双字长有符号整数 IN1 除以 IN2，产生一个字整数商 OUT，这里 IN1 与 OUT 是同一个存储单元	
完全整数乘法	MUL EN ENO ????-IN1 OUT-???? ????-IN2	MUL IN1，OUT	使能输入有效时，将两个字长的符号整数 IN1 和 IN2 相乘，产生一个双字整数结果 OUT，这里 IN2 与 OUT 的低 16 位是同一个存储单元	IN1/IN2 数据类型：整数；操作数：VW、MW、IW、QW、SW、SMW、LW、AC、*VD、*AC、*LD、T、C、常数 OUT 数据类型：双整数；VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD
完全整数除法	DIV EN ENO ????-IN1 OUT-???? ????-IN2	DIV IN2，OUT	使能输入有效时，用字型有符号整数 IN1 除以 IN2，产生一个双字整数结果 OUT，低 16 位存放商，高 16 位存放余数，这里 IN1 与 OUT 的低 16 位是同一个存储单元	

（续）

乘/除法指令	梯 形 图	指 令	功 能	数据类型及操作数
实数乘法	MUL_R EN ENO ????-IN1 OUT-???? ????-IN2	×R IN1，OUT	使能输入有效时，将两个字长的实数 IN1 和 IN2 相乘，产生一个实数结果 OUT，这里 IN2 与 OUT 是同一个存储单元	数据类型：实数 IN1/IN2 操作数：VD、MD、ID、QD、SD、SMD、LD、AC、＊VD、＊AC、＊LD、常数 OUT：VD、ID、QD、MD、SD、SMD、LD、AC、＊VD、＊AC、＊LD
实数除法	DIV_R EN ENO ????-IN1 OUT-???? ????-IN2	/R IN1，OUT	使能输入有效时，将双字长的实数 IN1 除以 IN2，产生一个实数结果 OUT，这里 IN1 与 OUT 是同一个存储单元	

二、指令练习

1. 增减指令练习

图 4.5-1 为增减指令的应用实例梯形图。图中 AC0 的初始值为 125，VD100 的初始值为 128000。当 I0.1 接通时，AC0 中的 125 加 1 后得到 126 存储到 AC0 中，VD100 中的 128000 减 1 后得到 127999 存储到 VD100 中。

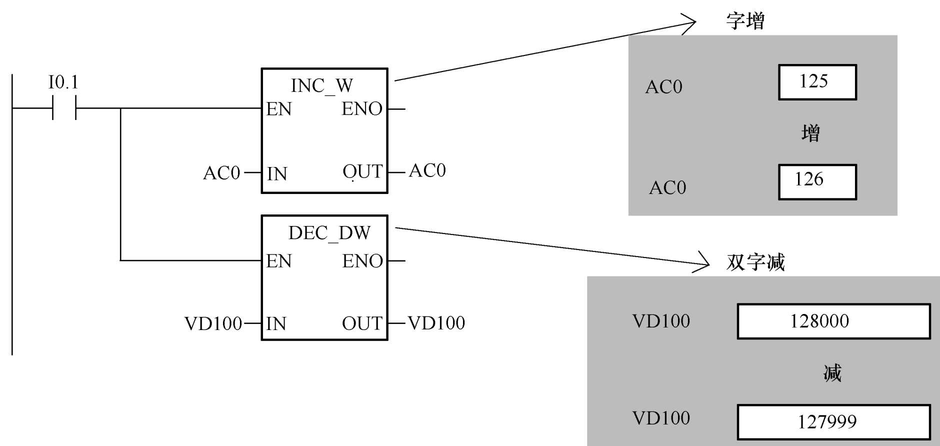


图 4.5-1 增减指令的应用梯形图

2. 整数加/减指令练习

整数的加、减、乘、除运算指令是将两个 16 位整数进行加、减、乘、除运算，产生一个 16 位的结果，而除法的余数不保留。如图 4.5-2 所示，当 I0.0 接通时，100 与 VW100 相加，结果存入 VW200；VW2 减去 1，结果存入 VW2。

3. 整数乘/除指令练习

如图 4.5-3 所示，初始时 AC1 中的内容为 40，VW100 中的内容为 20，VW200 中的内容为 4000，VW10 中的内容为 40。当 I0.0 接通时，AC1 乘以 VW100 等于 800 传送到 VW100 中，VW200 除以 VW10 等于 100 传送到 VW200 中，如图 4.5-4 所示。

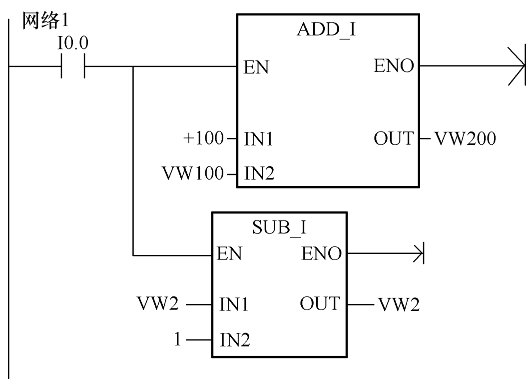


图 4.5-2　整数加/减运算程序示例

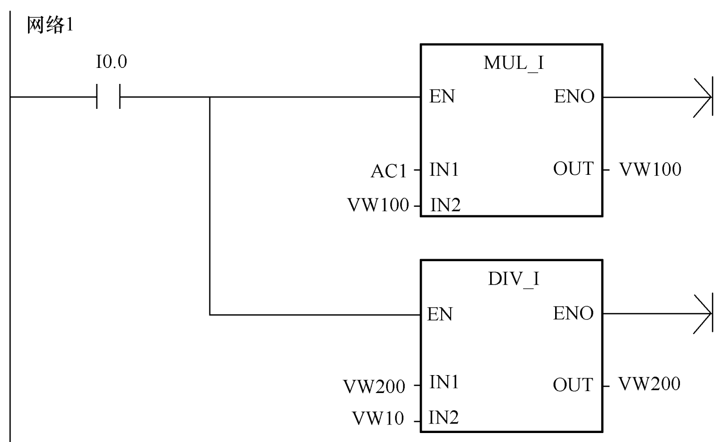


图 4.5-3　　梯形图

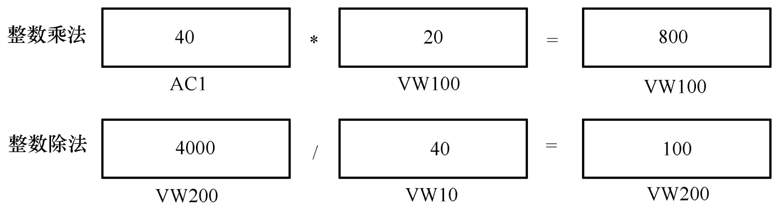


图 4.5-4　整数乘/除运算程序示例

第六节　跳 转 指 令

一、指令介绍

跳转指令可以根据不同的逻辑条件，有选择地执行不同的程序，可以使程序结构更加灵活，减少扫描时间，从而加快系统响应速度。跳转指令需要两条指令配合使用，即跳转开始指令（JMP）和跳转标号指令（LBL），具体指令介绍见表 4.6-1。

表 4.6-1　跳转指令表

指令类型	梯形图	指　令	指令功能	数据类型	操作数
跳转开始指令	n ——(JMP)	JMP　n	使能输入有效时，使程序跳转到指定标号 n 处执行；使能输入无效时，程序顺序执行	字	常数 0～255
跳转标号指令	n [LBL]	LBL　n	用来标记跳转指令的目的位置		

二、指令练习

图 4.6-1 所示为跳转指令应用的梯形图及语句表。当 I0.0 接通时，JMP 条件满足，程序跳转执行 LBL 以后的程序，在 JMP 和 LBL 之间的指令均不执行，即在此过程中，I0.1 即使接通，Q0.0 也不会接通，只有当 JMP 条件不满足时，I0.1 接通，Q0.0 才会接通。

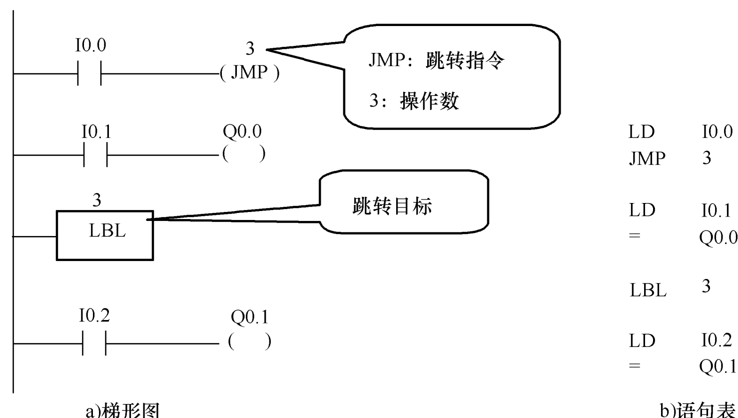


图 4.6-1　跳转指令的应用

使用说明：

1）跳转指令 JMP 和 LBL 必须配合应用，并且同时出现在同一程序中。不允许从主程序跳到子程序或中断程序，也不允许从子程序跳到主程序或中断程序。

2）在跳转条件中引入上升沿或下降沿脉冲指令时，跳转只能执行一个扫描周期。

3）在执行跳转后，Q、M、S、C 等元件的位保持跳转前的状态。定时器在跳转期间，分辨率 1ms 和 10ms 的定时器一直维持原工作状态，原来工作的会继续工作；对于分辨率 100ms 的定时器，在跳转期间停止工作，但不会复位，存储器里的值为跳转前的值，跳转结束后，若输入条件允许，可继续计时，但失去了准确计时的意义，所以在使用时要慎重。

第七节　转换指令

转换指令是指对操作数的类型进行转换，包括数据的类型转换、码的类型转换及数据和码之间的类型转换。

一、指令介绍

1. 数据类型转换指令

数据类型转换指令包括字节与整数之间的转换、整数与双整数之间的转换以及双整数与实数之间的转换指令，具体指令介绍见表 4. 7-1。

表 4. 7-1　数据类型转换指令表

指令类型	梯形图	指令	功能	数据类型及操作数
字节到整数	I-B：EN ENO；????-IN OUT-????	BTI IN，OUT	输入端 EN 有效时，将输入（IN）的字节型数据转换成整数，并存入 OUT	IN 数据类型：字节；操作数：VB、IB、QB、MB、SB、SMB、LB、AC、常量、*VD、*AC、*LD OUT 数据类型：整数；操作数：VW、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC
整数到字节	B-I：EN ENO；????-IN OUT-????	ITB IN，OUT	输入端 EN 有效时，将输入（IN）的整数据转换成字节型数据，并存入 OUT	IN 数据类型：整数；操作数：VB、IB、QB、MB、SB、SMB、LB、AC、常量、*VD、*AC、*LD OUT 数据类型：字节；操作数：VW、IW、QW、MW、SW、SMW、LW、T、C、AC、*VD、*LD、*AC
整数到双整数	I-DI：EN ENO；????-IN OUT-????	ITD IN，OUT	输入端 EN 有效时，将输入（IN）的整数转换成双整数，并存入 OUT	IN 数据类型：整数；操作数：VB、IB、QB、MB、SB、SMB、LB、AC、常量、*VD、*AC、*LD OUT 数据类型：双整数；操作数：VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD
双整数到整数	DI-I：EN ENO；????-IN OUT-????	DTI IN，OUT	输入端（IN）的有符号双整数转换成整数，并存入OUT	IN 数据类型：双整数；操作数：VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD OUT 数据类型：整数；操作数：VB、IB、QB、MB、SB、SMB、LB、AC、常量、*VD、*AC、*LD
实数到双整数	ROUND：EN ENO；????-IN OUT-????	ROUND IN，OUT	ROUND 取整指令，转换时实数的小数部分四舍五入	IN 数据类型：实数；操作数：VD、ID、QD、MD、SD、SMD、LD、AC、*VD、*AC、*LD OUT 数据类型：双整数；操作数：VD、ID、QD、MD、SD、SMD、LD、AC、HC、*VD、*AC、*LD
实数到双整数	TRUNC：EN ENO；????-IN OUT-????	TRUNC IN，OUT	TRUNC 取整指令，实数舍去小数部分后，转换成 32 位有符号整数	

（续）

指令类型	梯形图	指令	功能	数据类型及操作数
双整数到实数	DI-R EN ENO ????-IN OUT-????	DTR IN，OUT	将输入端（IN）指定的32位有符号整数转换成32位实数	IN数据类型：双整数；操作数：VD、ID、QD、MD、SD、SMD、LD、AC、HC、＊VD、＊AC、＊LD OUT数据类型：实数；操作数：VD、ID、QD、MD、SD、SMD、LD、AC、＊VD、＊AC、＊LD
提示：如果想将一个整数转换成实数，则先用整数转双整数（ITD）指令，再用双整数转实数（DTR）指令。				

2. BCD 码转换指令

BCD码转换指令包括BCD码转换成整数（BCDI）指令和整数转换成BCD码（IBCD）指令，具体指令介绍见表4.7-2。

表4.7-2　BCD码转换指令表

指令类型	梯形图	指令	功能	数据类型及操作数
整数到BCD码	I_BCD EN ENO ????-IN OUT-????	IBCD　OUT	将输入整数值IN转换成二进制编码的十进制数，并将结果载入OUT指定的变量中	数据类型：字 IN操作数：VW、IW、QW、MW、SW、SMW、LW、T、C、AIW、AC、常量、＊VD、＊AC、＊LD OUT：VW、IW、QW、MW、SW、SMW、LW、T、C、AC、＊VD、＊LD、＊AC
BCD码到整数	BCD_I EN ENO ????-IN OUT-????	BCDI　OUT	将二进制编码的十进制数值IN转换成整数，并将结果载入OUT指定的变量中。IN的有效范围是0～9999BCD	

3. 七段码指令

在S7-200 PLC中，有一条可直接驱动七段数码管的指令SEG，该指令在数码管的显示中直接应用非常方便，具体指令介绍见表4.7-3，七段码编码见表4.7-4。

表4.7-3　译码指令表

指令名称	梯形图	指令	功能	数据类型及操作数
七段显示译码指令	SEG EN ENO IN OUT	SEG IN，OUT	当输入端EN有效时，把输入字节（IN）低4位确定的有效十六进制数（16#0～F）产生点亮七段显示器各段的代码（七段显示码），并送到输出OUT字节单元	数据类型：字节 IN操作数：VB、IB、QB、MB、SB、SMB、LB、AC、常量、＊VD、＊AC、＊LD OUT：VB、IB、QB、MB、SMB、LB、AC、＊VD、＊AC、SB、＊LD

表 4.7-4　七段码编码表

输入 LSD	七段码显示器	输出 －gfe　dcba		输入 LSD	七段码显示器	输出 －gfe　dcba
0	0	0011　1111	a f g b e c d	8	8	0111　1111
1	1	0000　0110		9	9	0110　0111
2	2	0101　1011		A	A	0111　0111
3	3	0100　1111		B	b	0111　1100
4	4	0110　0110		C	C	0011　1001
5	5	0110　1101		D	d	0101　1110
6	6	0111　1101		E	E	0111　1001
7	7	0000　0111		F	F	0111　0001

4. 译码和编码指令

译码 DECO（Decode）指令、编码 ENCO（Encode）指令的介绍见表 4.7-5。

表 4.7-5　DECO、ENCO 指令表

指令名称	梯形图	语句表	功能	操作数及数据类型
DECO 指令	DECO EN　ENO IN　OUT	DECO IN，OUT	译码指令根据输入字节（IN）的低 4 位表示的输出字的位号，将输出字的相对应位置位为 1，输出字的其他位均置位为 0	IN：VB、IB、QB、MB、SMB、LB、SB、AC、常量；数据类型：字节 OUT：VW、IW、QW、MW、SMW、LW、SW、AQW、T、C、AC；数据类型：字
ENCO 指令	ENCO EN　ENO IN　OUT	ENCO IN，OUT	编码指令将输入字（IN）最低有效位（其值为 1）的位号写入输出字节（OUT）的低 4 位中	IN：VW、IW、QW、MW、SMW、LW、SW、AIW、T、C、AC、常量；数据类型：字 OUT：VB、IB、QB、MB、SMB、LB、SB、AC；数据类型：字节

二、指令练习

1. BCD 码到整数转换指令练习

图 4.7-1 所示为四位 8421BCD 拨码器。8421BCD 拨码器能够将由按键输入的 1 位十进制数（0～9）以 BCD 码的形式输出。

拨码器由处于前面板的拨码盘和处于后侧板的接线端组成。拨码盘由上下两个拨盘按钮和夹在按钮中间的拨位数码指示器组成。拨位数码指示器可随拨码盘的拨动而转动 0～9 十个数字，用以显示拨码盘当前数值。上面的拨码按钮为增量按钮，每按下一次，拨码盘正向旋转 1/10 周，拨位数码指示器显示的数值加 1，连续按十次，数据将被还原；下面的拨码按钮为减量按钮，每按下一次，拨码器反向旋转 1/10 周，拨位数码指示器显示的数值减 1。

接线端向外引出标有 8、4、2、1、A 的五个引脚。在实际应用中，BCD 拨码器可以直接插入 BCD 拨码器插座中使用，也可以采取从五个引脚上分别焊接引线的方式使用。

BCD 拨码器接线端的通断状态是当前拨码器位置的反映，拨码盘数码显示的数值直接影响 8、4、2、1 这四个引脚与公共引脚 A 的导通状态。例如，当前拨码盘拨位数码指示器的显示数据为 7 时，图 4.7-1 中的 4、2、1 引脚均与 A 导通，8 引脚与 A 不导通。拨码盘从 0 拨到 9，A 引脚与 8、4、2、1 四个引脚的导通状态见表 4.7-6。此表中的 0 表示输入控制线 A 与输出线不通，表中的 1 表示输入控制线 A 与输出线导通。

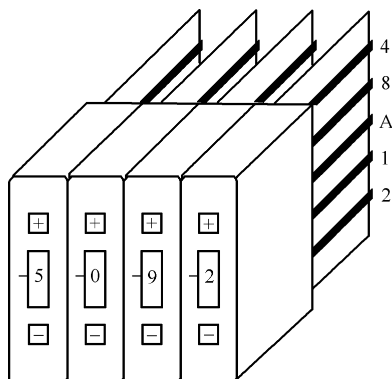


图 4.7-1　四位 8421BCD 拨码器

表 4.7-6　BCD 拨码器状态表

位　置	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

有两个两位 BCD 拨码器，其中一个拨码器的 8、4、2、1 引脚分别与 I0.7、I0.6、I0.5、I0.4 连接，组成十进制数的十位；另一个拨码器的 8、4、2、1 引脚分别与 I0.3、I0.2、I0.1、I0.0 连接，组成十进制数的个位。当拨码器拨入十进制数 95 时，先将 95 变为 BCD 码 1001 0101，然后存入 VB1，再将 VW0 中的 BCD 码 0000 0000 1001 0101 转换为整数（即二进制编码的十进制数值）输出到 QW0，如图 4.7-2 所示。

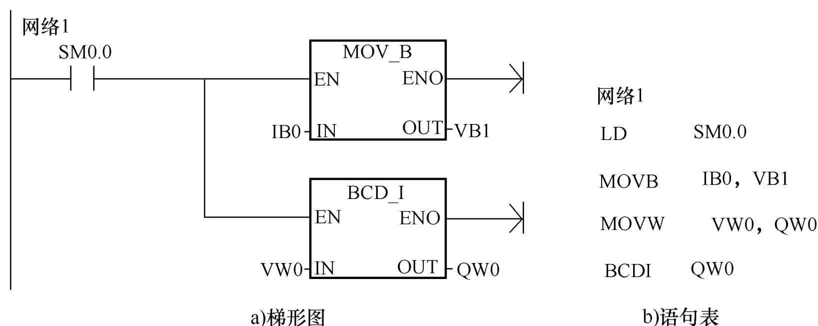


图 4.7-2　BCDI 指令应用举例

由图4.7-3所示的工作过程可以看出，VW0中存储的BCD码与QW0中存储的二进制数据完全不同。VW0以4位BCD码为1组，从高至低分别是十进数0、0、9、5的BCD码。

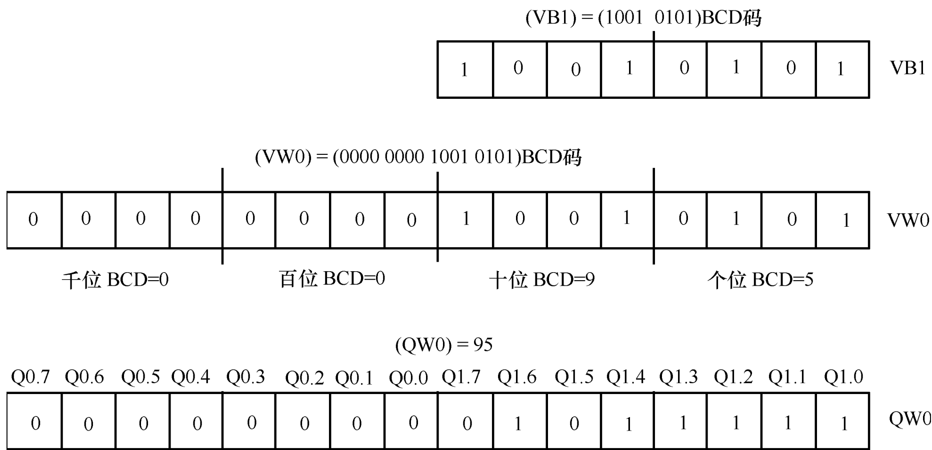


图4.7-3　BCDI指令工作过程

2. 七段码指令练习

图4.7-4为七段数码管的控制梯形图。当I0.0闭合时上升沿脉冲使计数器C0计数，并传送到VW0中。七段显示译码指令把VW0的低4位（VB1）二进制数码转换成七段显示码输出到QB0中，驱动数码管显示数字。

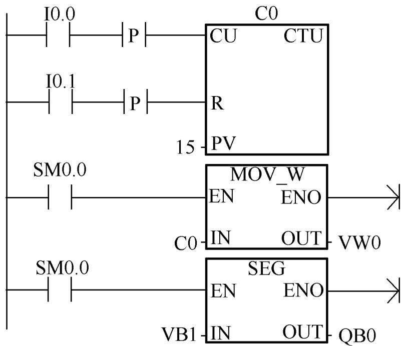
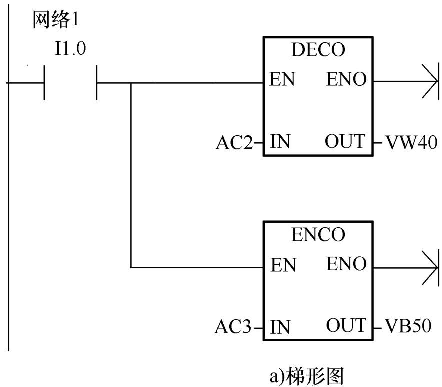


图4.7-4　七段数码管的控制梯形图

3. 译码、编码指令练习

译码与编码指令的应用如图4.7-5所示。若AC2 = 2，执行译码指令，则将输出字VW40的第二位置1，VW40中的二进制数为2#0000 0000 0000 0010；若AC3 = 2#0000 0000 0000 0010，执行编码指令，则输出字节VB50中的码为2。



a)梯形图

网络1

LD	I1.0
DECO	AC2，VW40
ENCO	AC3，VB50

b)语句表

图4.7-5　译码与编码指令的应用

第八节　子程序与中断指令

一、指令介绍

1. 子程序指令

（1）子程序的作用　通常将具有特定功能并且多次使用的程序段作为子程序，别的程序在需要子程序时就可以调用它，而无需重写。子程序的调用是有条件的，未调用它时不会执行子程序的指令，因此使用子程序可以减少扫描时间。使用子程序还可以将程序分成容易管理的小块，使程序结构简单清晰，易于查错和维护。

如果子程序中只引用参数和局部变量，因为与其他 POU 没有地址冲突，则可以将子程序移植到其他工程项目。为了移植子程序，应避免使用全局符号和变量，如 I、Q、M、SM、AI、AQ、V、T、C、S、AC 等存储器中的绝对地址。

欲在程序中使用子程序，必须执行下列三项任务：

1）建立子程序。

2）在子程序局部变量表中定义参数（带参数调用子程序时必须执行）。

3）从适当的 POU（从主程序或另一个子程序）调用子程序。

（2）建立子程序的方法　建立子程序最简单的方法是在程序编辑器中的空白处单击鼠标右键，再选择“插入”→“子程序”命令即可，如图 4.8-1 所示。

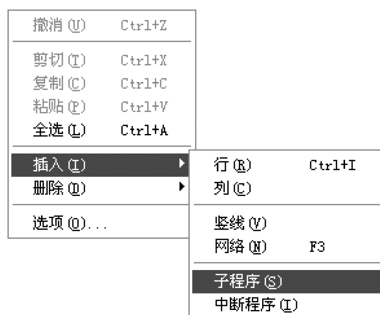


图 4.8-1　建立子程序最简单的方法

也可以选择菜单栏“编辑”→“插入”→“子程序”命令，或者在指令树中，用鼠标右键单击“程序块”图标，从弹出的菜单中选择“插入”→“子程序”命令建立子程序。程序编辑器将从原来的 POU 显示进入新的子程序，程序编辑器底部出现标志新的子程序的新标签，在程序编辑器窗口中可以对新的子程序编程。用鼠标右键单击指令树中的子程序的图标或程序编辑器底部的子程序的标签，在弹出的菜单中选择“重命名”，可修改它们的名称，也可以通过双击程序编辑器底部的子程序的标签直接修改子程序的名称。

（3）子程序指令介绍　S7-200 PLC 的指令系统中具有简单、方便、灵活的子程序调用指令，在结构化程序设计中是一种方便有效的工具。子程序指令包括子程序调用、子程序有条件返回和带参数的子程序调用，指令介绍见表 4.8-1。

（4）子程序的调用　可以在主程序、另一子程序或中断程序中调用子程序，但是不能在子程序中调用自己（不允许递归调用），也可以带参数或不带参数调用子程序。

调用子程序时将执行子程序的全部指令，直至子程序结束，然后返回调用程序中子程序调用指令的下一条指令之处。

在梯形图程序中插入子程序调用指令时，首先打开程序编辑器窗口中需要调用子程序的 POU，找到需要调用子程序的地方；然后在指令树的最下面用左键打开子程序文件夹，将需要调用的子程序图标从指令树拖到程序编辑器中的正确位置，或将光标置于程序编辑器窗口

中双击指令树中的调用指令。

表 4.8-1　子程序指令表

指令名称	梯形图	语句表	说明
子程序调用指令	SBR_n —EN	CALL　SBR_n	① 子程序调用指令编在主程序中，子程序返回指令编在子程序中，子程序的标号 n 的范围是0 ~ 63 ② 子程序返回分为有条件返回和无条件返回，无条件子程序返回指令 RET 为自动默认；有条件子程序返回指令 CRET，条件成立时结束该子程序，返回到原调用处的下一条指令开始执行
子程序有条件返回指令	——(RET)	CRET	

在语句表程序中调用子程序时使用调用指令“SBR_n”。在调用子程序时，CPU 保存整个逻辑堆栈后，将栈顶值置为 1，堆栈中的其他值清 0，控制转移至被调用的子程序。子程序执行完成后，用调用时保存的数据恢复堆栈，控制返回调用程序。子程序和调用程序共用累加器，不会因使用子程序自动保存或恢复累加器。

（5）子程序的嵌套调用　程序中最多可创建 64 个子程序。子程序可以嵌套调用（在子程序中调用别的子程序），最大嵌套深度为 80。

（6）程序的有条件返回　在程序中，用触点电路控制 RET（从子程序有条件返回）指令，触点电路接通时条件满足，子程序被终止。编程软件自动地为主程序和子程序添加无条件返回指令。

类似地，可以在主程序中用触点电路驱动 END（有条件结束）指令。

（7）子程序中的定时器　停止调用子程序时，在子程序内的定时器线圈的 ON/OFF 状态保持不变。如果在停止调用时子程序中的定时器正在计时，100ms 定时器将停止计时，当前值保持不变，重新调用时继续计时；但是 1ms 定时器和 10ms 定时器将继续计时，定时时间到，它们的定时器位变为 1 状态，并且可以在子程序之外起作用。

2. 中断指令

所谓中断就是当 CPU 执行正常程序时，系统中出现了某些急需处理的特殊请求，这时 CPU 暂时中断正在执行的程序，转而去对随机发生的更加紧急的事件进行处理（称为执行中断服务程序）。当该事件处理完毕后，CPU 自动返回原来被中断的程序继续执行。执行中断服务程序前后，系统会自动保护被中断程序的运行环境，故不会造成混乱。

（1）中断源及种类　中断源即中断事件发出中断请求的来源。S7-200 PLC 具有最多可达 34 个中断源，每个中断源都分配一个编号用以识别，称为中断事件号。这些中断源大致分为三大类：通信口中断、输入/输出中断和时基中断。

1）通信口中断。通信口中断包括通信口 0 和通信口 1 产生的中断。S7-200 的串行通信口可以由用户程序来控制。用户可以通过编程的方法来设置通信协议、波特率和奇偶校验等参数。对通信口的这种操作方式，又称为自由口通信。利用 PLC 接收、发送字符可以产生中断事件，以简化程序对通信的控制。S7-200 PLC 有 6 种通信口中断事件。

2）输入/输出（I/O）中断。S7-200 对 I/O 点状态的各种变化产生中断，包括外部输入中断（上升沿中断或下降沿中断）、高速计数器 HC 中断和脉冲串输出（PTO）中断。这些事件可以对输入的上升或下降状态、高速计数器或者脉冲串输出做出响应。

外部输入中断是系统利用 I0.0 ~ I0.3 的上升沿或下降沿产生中断，这些输入点可用于连接某些一旦发生必须引起注意的外部事件；高速计数器中断可以响应当前值等于预设值、计数方向改变、计数器外部复位等事件引起的中断，高速计数器的中断可以实时得到迅速响应，从而实现比 PLC 扫描周期还要短的控制任务；脉冲串输出中断用来响应给定数量脉冲输出完成引起的中断，脉冲串输出主要的应用是步进电动机。

3）时基中断。时基中断包括定时中断（Timed Interrupt）和定时器 T32、T96 中断。

① 定时中断用来执行一个周期性的操作。周期时间以 ms 为单位，周期时间范围为 1 ~ 255ms。对于定时中断 0，把周期时间值写入 SMB34；对于定时中断 1，把周期时间值写入 SMB35。当达到设定周期时间值时，定时器溢出，执行中断处理程序。如果定时中断事件已被连接到一个定时中断程序，为了改变定时中断的时间间隔，首先必须修改 SMB34 或 SMB35 的值，然后重新把中断程序连接到定时中断事件上。如果退出 RUN 状态或者定时中断被分离，则定时中断被禁止。通常用定时中断以固定的时间间隔去控制模拟量输入的采样或者执行一个 PID 回路。

② 定时器中断是利用定时器对一个指定的时间段产生中断。这类中断只能使用 1ms 的定时器 T32 和 T96。当 T32 或 T96 的当前值等于预置值时，CPU 响应定时器中断，执行中断服务程序。

（2）中断优先级　在 PLC 应用系统中通常有多个中断事件。当多个中断事件同时向 CPU 申请中断时，要求 CPU 能够将全部中断事件按中断性质和轻重缓急进行排队，并依优先权高低逐个处理。

S7-200 CPU 规定的中断优先权由高到低依次是通信中断、I/O 中断和定时中断，每类中断又有不同的优先级。

在上述三个优先级范围内，CPU 按照先来先服务的原则处理中断，任何时刻只能执行一个用户中断程序。一旦一个中断程序开始执行，它要一直执行到完成，即使另一个中断程序的优先级较高，也不能中断正在执行的中断程序。正在处理其他中断时发生的中断事件则排队等待处理。如果中断事件的产生过于频繁，使中断产生的速率比可以处理的速率快，或者中断被 DISI 指令禁止，中断队列溢出状态位被置 1。只应在中断程序中使用这些位，因为当队列变空或返回主程序时这些位被复位。主机中的所有中断事件及优先级见表 4.8-2。

表 4.8-2　中断事件及优先级表

中 断 号	中断事件描述	优先级分组	组 内 类 型	组中优先级
8	通信端口 0：接收字符	通信（最高）	通信端口 0	0
9	通信端口 0：发送完成			0
23	通信端口 0：接收信息完成			0
24	通信端口 1：接收信息完成		通信端口 1	1
25	通信端口 1：接收字符			1
26	通信端口 1：发送完成			1
19	PTO 0 脉冲串输出完成中断	I/O（中等）	脉冲串输出	0
20	PTO 1 脉冲串输出完成中断			1

（续）

中 断 号	中断事件描述	优先级分组	组 内 类 型	组中优先级
0	I0.0 上升沿中断	I/O（中等）	外部输入	2
2	I0.1 上升沿中断			3
4	I0.2 上升沿中断			4
6	I0.3 上升沿中断			5
1	I0.0 下降沿中断			6
3	I0.1 下降沿中断			7
5	I0.2 下降沿中断			8
7	I0.3 下降沿中断			9
12	HC0 CV = PV（当前值 = 预置值）中断			10
27	HC0 计数方向改变中断			11
28	HC0 外部信号复位中断			12
13	HC1 CV = PV（当前值 = 预置值）中断			13
14	HC1 计数方向改变中断			14
15	HC1 外部信号复位中断			15
16	HC2 CV = PV（当前值 = 预置值）中断		高速计数器	16
17	HC2 计数方向改变中断			17
18	HC2 外部信号复位中断			18
32	HC3 CV = PV（当前值 = 预置值）中断			19
29	HC4 CV = PV（当前值 = 预置值）中断			20
30	HC4 计数方向改变中断			21
31	HC4 外部信号复位中断			22
33	HC5 CV = PV（当前值 = 预置值）中断			23
10	定时中断 0，SMB34	定时（最低）	定时	0
11	定时中断 1，SMB35			1
21	定时器 T32　CT = PT 中断		定时器	2
22	定时器 T96　CT = PT 中断			3

（3）中断指令介绍　中断指令共有 6 条，包括中断连接、中断分离、清除中断事件、中断禁止、中断允许和中断条件返回，具体指令介绍见表 4.8-3。

（4）中断程序　中断程序不是由程序调用，而是在中断事件发生时由操作系统调用，使系统对特殊的内部或外部事件做出响应。系统响应中断时自动保存逻辑堆栈、累加器和某些特殊标志存储器位，即保护现场；中断处理完成时，又自动恢复这些单元原来的状态，即恢复现场。因为不能预知系统何时调用中断程序，在中断程序中不能改写其他程序使用的存储器，为此应在中断程序中尽量使用局部变量，并妥善分配各 POU 使用的全局变量，保证中断程序不会破坏别的 POU 使用的全局变量中的数据。在中断程序中可以调用一级子程序，累加器和逻辑堆栈在中断程序和被调用的子程序中是公用的。

表 4.8-3　中断指令的梯形图和语句表

指令名称	梯形图	指令	功能	数据类型及操作数
中断连接指令	ATCH EN ENO ???? IN1 ???? EVNT	ATCH IN EVNT	将一个中断事件和一个中断程序建立联系，并允许这一中断事件	中断程序号 INT 和中断事件号 EVNT 均为字节型常数。 EVNT 的数值：CPU221 和 CPU222 的 EVNT 取值范围为 0 ~ 12，19 ~ 23，27 ~ 33；CPU224 的 EVNT 取值范围为 0 ~ 23，27 ~ 33；CPU226 和 CPU226XM 的 EVNT 取值范围为 0 ~ 33
中断分离指令	DTCH EN ENO ???? EVNT	DTCH EVNT	切断一个中断事件和所有程序的联系，使该事件的中断回到不激活或无效状态，因而禁止了该中断事件	
开中断指令	—(ENI)	ENI	允许全局开放所有被连接的中断事件	无操作数
关中断指令	—(DISI)	DISI	全局关闭所有连接的中断事件	无操作数

中断处理提供对特殊内部事件或外部事件的快速响应。中断程序越短越好，减少中断程序的执行时间，避免引起主程序控制的设备操作异常。

在编写中断程序前，先创建中断程序，创建成功后将显示新的中断程序的标签。建立中断程序的方法有如下几种。

方法一：选择菜单栏“编辑”→“插入”→“中断”命令。

方法二：用鼠标右键单击指令树中的“程序块”图标并从弹出的菜单中选择“插入”→“中断”命令。

方法三：在程序编辑器窗口单击右键，从弹出的菜单中单击“插入”→“中断”命令。

程序编辑器从先前的 POU 显示更改为新中断程序，在程序编辑器的底部会出现一个新标记，代表新的中断程序。

（5）使用中断的注意事项

1）在启动中断程序之前，应在中断事件和该事件发生时希望执行的中断程序之间，用 ATCH 指令建立联系。执行 ATCH 指令后，该中断程序在事件发生时被自动启动。

2）程序开始运行时，CPU 默认禁止所有中断。如果执行了中断允许指令 ENI，则允许所有中断。

3）一个事件只能连接一个中断程序，而多个中断事件可以调用同一个中断程序，但是一个中断事件不能同时调用多个中断程序。中断被允许且中断事件发生时，将执行为该事件指定的最后一个中断程序。

4）执行中断分离指令 DTCH 时，只禁止某个事件与中断程序的联系；而执行中断禁止指令 DISI 时，则禁止所有中断。

5）在中断程序中不能使用 DISI、ENI、HDFE、FOR/NEXT、LSCR 和 END 等指令。

6）程序中有多个中断子程序时，要分别编号。在建立中断程序时，系统会自动编号，也可以更改编号。

二、指令练习

1. 子程序指令练习

设计子程序：当 I0.0 闭合时，执行手动程序；I0.0 断开时，执行自动程序。主程序、子程序 SBR_0、子程序 SBR_1 分别如图 4.8-2a、b、c 所示。

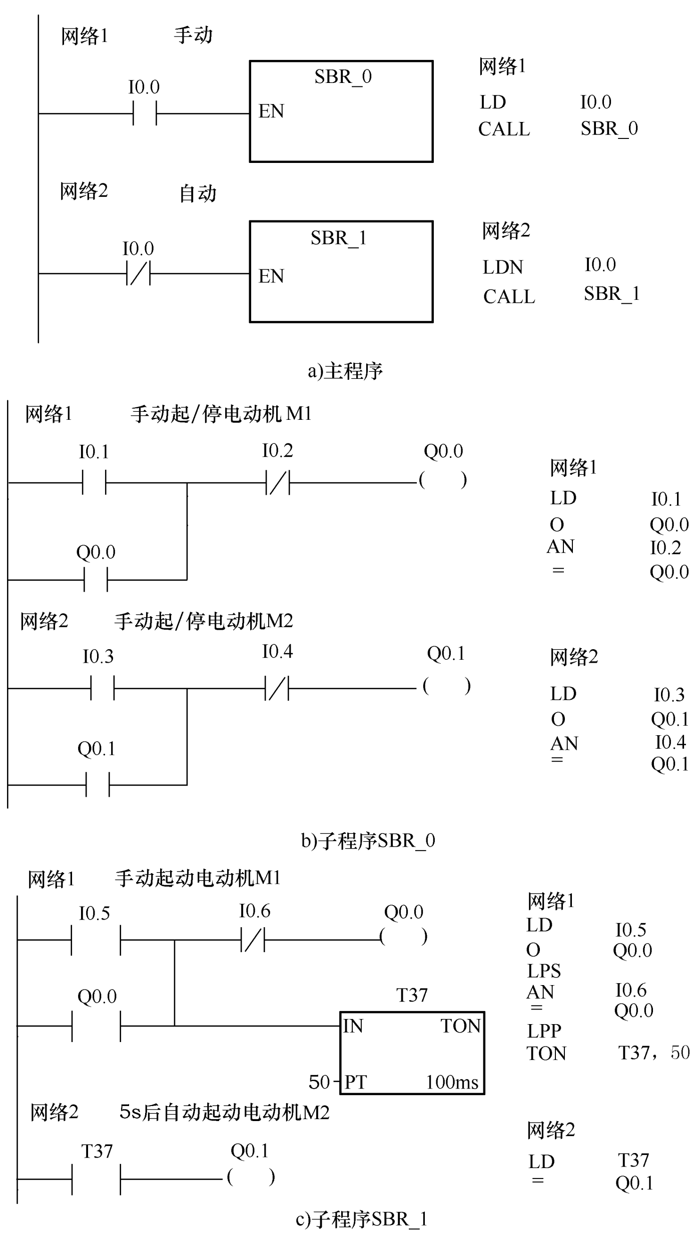


图 4.8-2　子程序指令应用示例

2. 中断指令练习

练习 1：I/O 中断的练习。在 I0.0 的上升沿通过中断使 Q0.0 立即置位，在 I0.1 的下降沿通过中断使 Q0.0 立即复位，程序设计如图 4.8-3 所示。

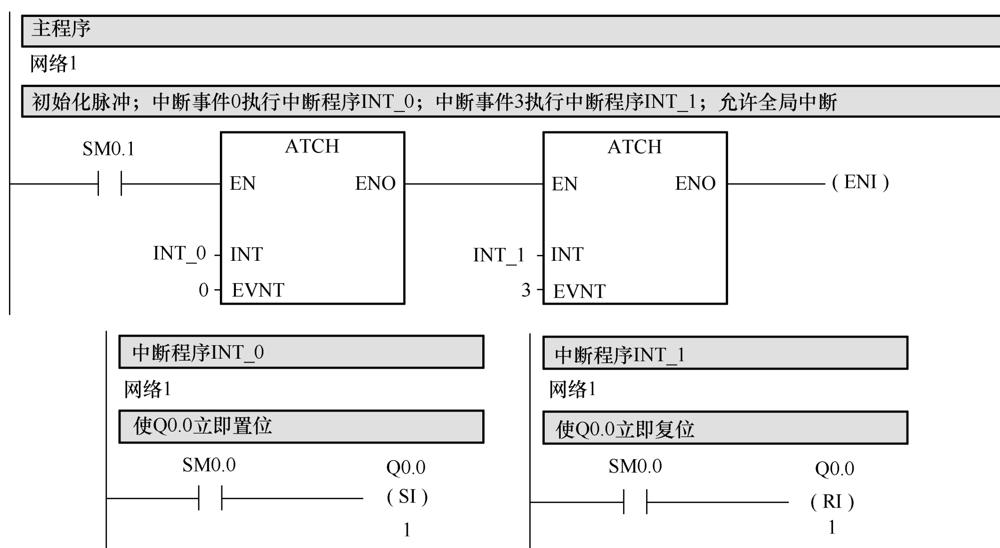
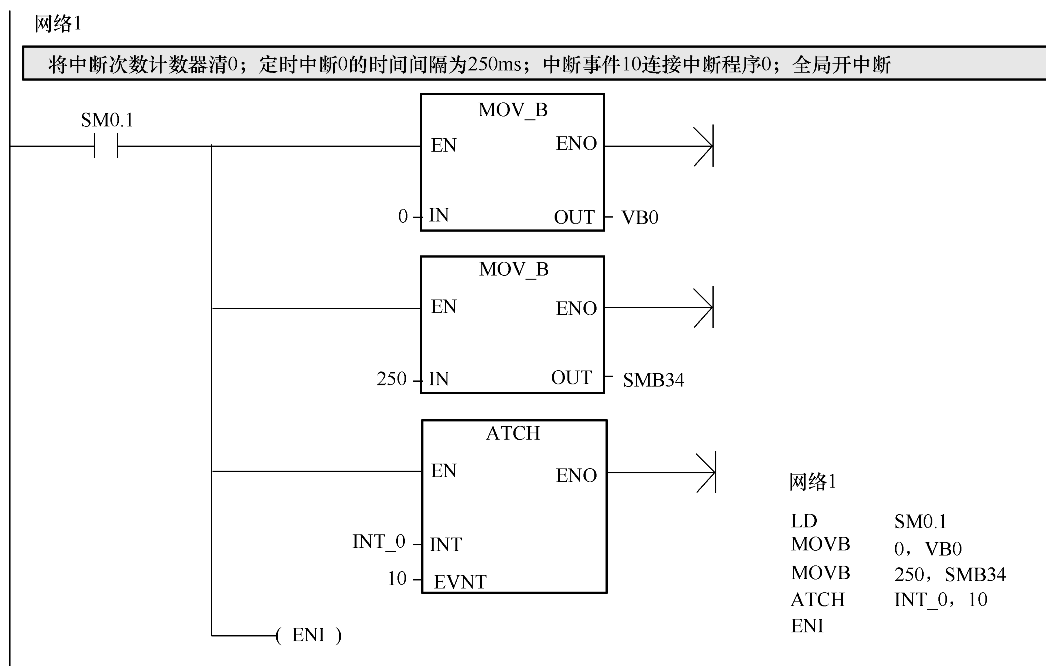


图 4. 8-3　I/O 中断的应用梯形图

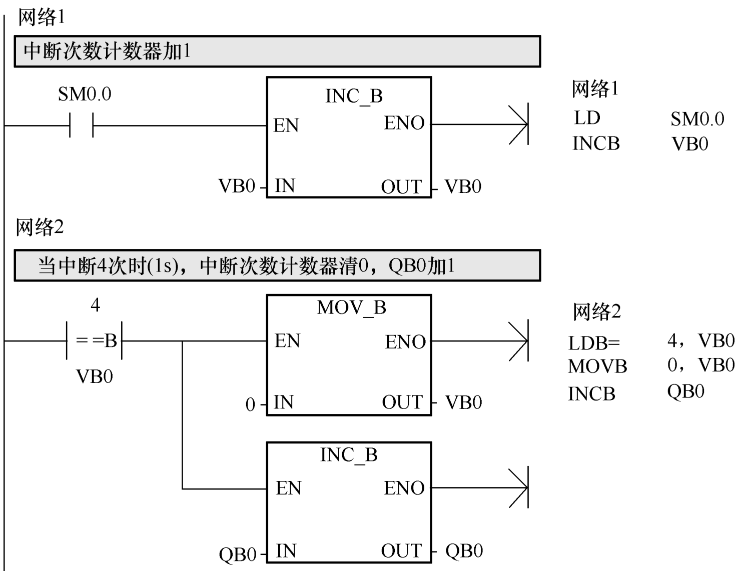
练习 2：定时中断的练习。用定时中断 0 实现周期为 1s 的高精度定时，并在 QB0 端口以增 1 形式输出。

为了实现周期为 1s 的高精度周期性操作的定时，将定时中断的定时时间间隔设为 250ms，在定时中断 0 的中断程序中，将 VB0 加 1，然后用比较触点指令“LD =”判断 VB0 是否等于 4，若相等（中断了 4 次，对应的时间间隔为 1s），在中断程序中执行每 1s 一次的高精度周期性操作的定时，程序设计如图 4. 8-4 所示。



a)主程序

图 4. 8-4　定时中断的应用



b)中断程序INT_0

图 4.8-4　定时中断的应用（续）

第九节　逻辑运算指令

一、指令介绍

逻辑运算是对无符号数进行的逻辑处理，主要包括逻辑与、逻辑或、逻辑异或和取反等运算指令。

1. 取反指令 INV

取反指令有字节、字、双字取反指令，具体介绍见表 4.9-1。

表 4.9-1　取反指令表

指令名称	梯形图	语句表	功能	操作数及数据类型
字节取反指令	INV_B EN ENO IN OUT	INVB　OUT	使能输入有效时，将一个字节长的输入逻辑数按位取反，得到一个字节逻辑运算结果，送到 OUT 指定的存储器单元输出	IN：VB、IB、QB、MB、SB、SMB、LB、AC、常数、* VD、* AC、* LD；数据类型：字节 OUT：VB、IB、QB、MB、SB、SMB、LB、AC、* VD、* AC、* LD；数据类型：字节
字取反指令	INV_W EN ENO IN OUT	INVW　OUT	使能输入有效时，把一个字长的输入逻辑数按位取反，得到一个字逻辑运算结果，送到 OUT 指定的存储器单元输出	IN：VW、IW、QW、MW、SW、SMW、T、C、AC、LW、AIW、常数、* VD、* AC、* LD；数据类型：字 OUT：VW、IW、QW、MW、SW、SMW、T、C、AC、LW、* VD、* AC、* LD；数据类型：字

（续）

指令名称	梯形图	语句表	功能	操作数及数据类型
双字取反指令	INV_DW EN　ENO IN　OUT	INVD　OUT	使能输入有效时，把一个双字长的输入逻辑数按位取反，得到一个双字长逻辑运算结果，送到 OUT 指定的存储器单元输出	IN：VD、ID、QD、MD、SD、SMD、LD、常数、HC、AC、＊VD、＊LD、＊AC；数据类型：双字 OUT：VD、ID、QD、MD、SD、SMD、LD、AC、＊VD、＊LD、＊AC；数据类型：双字

2. 逻辑与、或和异或指令（见表 4.9-2）

表 4.9-2　逻辑与、或和异或指令表

逻辑运算指令	梯形图	指令	指令功能	数据类型及操作数
字节与	WAND_B EN　ENO ????-IN1　OUT-???? ????-IN2	ANDB IN1，OUT	使能输入有效时，把一个字节长的 IN1 和 IN2 的输入逻辑数按位相与，得到一个字节逻辑运算结果，送到 OUT 指定的存储器单元输出	数据类型：字节 IN1/IN2/N 操作数：VB、MB、IB、QB、SB、SMB、LB、AC、＊VD、＊AC、＊LD、常数 OUT：VB、IB、QB、MB、SMB、LB、AC、＊VD、＊AC、＊LD
字节或	WOR_B EN　ENO ????-IN1　OUT-???? ????-IN2	ORB IN1，OUT	使能输入有效时，将一个字节长的 IN1 和 IN2 的输入逻辑数按位相或，得到一个字节逻辑运算结果，送到 OUT 指定的存储器单元输出	
字节异或	WXOR_B EN　ENO ????-IN1　OUT-???? ????-IN2	XORB IN1，OUT	使能输入有效时，将一个字节长的 IN1 和 IN2 的输入逻辑数按位异或，得到一个字节逻辑运算结果，送到 OUT 指定的存储器单元输出	
字与	WAND_W EN　ENO ????-IN1　OUT-???? ????-IN2	ANDW IN1，OUT	使能输入有效时，把一个字长的 IN1 和 IN2 的输入逻辑数按位相与，得到一个字逻辑运算结果，送到 OUT 指定的存储器单元输出	数据类型：字 IN1/IN2/N 操作数：VW、MW、IW、QW、SW、SMW、LW、AC、＊VD、＊AC、＊LD、T、C、常数 OUT：VW、IW、QW、MW、SW、SMW、LW、AC、＊VD、＊AC、＊LD、T、C
字或	WOR_W EN　ENO ????-IN1　OUT-???? ????-IN2	ORW IN1，OUT	使能输入有效时，把一个字长的 IN1 和 IN2 的输入逻辑数按位相或，得到一个字逻辑运算结果，送到 OUT 指定的存储器单元输出	
字异或	WXOR_W EN　ENO ????-IN1　OUT-???? ????-IN2	XORW IN1，OUT	使能输入有效时，把一个字长的 IN1 和 IN2 的输入逻辑数按位异或，得到一个字逻辑运算结果，送到 OUT 指定的存储器单元输出	

（续）

逻辑运算指令	梯形图	指令	指令功能	数据类型及操作数
双字与	WAND_D EN ENO ???? IN1 OUT ???? ???? IN2	ANDD IN1，OUT	使能输入有效时，把一个双字长的 IN1 和 IN2 的输入逻辑数按位相与，得到两个双字长逻辑运算结果，送到 OUT 指定的存储器单元输出	数据类型：双字 IN1/IN2/N 操作数：VD、MD、ID、QD、SMD、LD、AC、HC、＊VD、＊AC、＊LD、常数 OUT：VD、ID、QD、MD、SMD、LD、AC、＊VD、＊AC、＊LD
双字或	WOR_D EN ENO ???? IN1 OUT ???? ???? IN2	ORD IN1，OUT	使能输入有效时，把一个双字长的 IN1 和 IN2 的输入逻辑数按位相或，得到两个字长逻辑运算结果，送到 OUT 指定的存储器单元输出	
双字异或	WXOR_D EN ENO ???? IN1 OUT ???? ???? IN2	XORD IN1，OUT	使能输入有效时，把两个双字长的 IN1 和 IN2 的输入逻辑数按位相异或，得到的两个字长逻辑运算结果，送到 OUT 指定的存储器单元输出	

二、指令练习

图 4.9-1 为字节与、字节或、字节异或指令的应用实例梯形图。当 I0.0 接通时，其运行过程如图 4.9-2 所示。

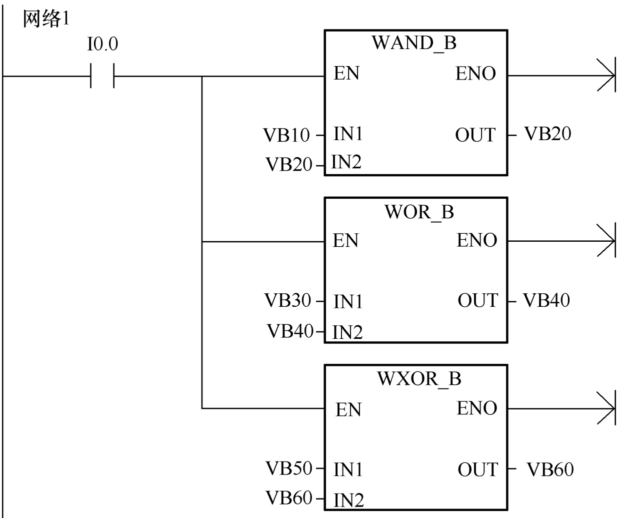


图 4.9-1　梯形图

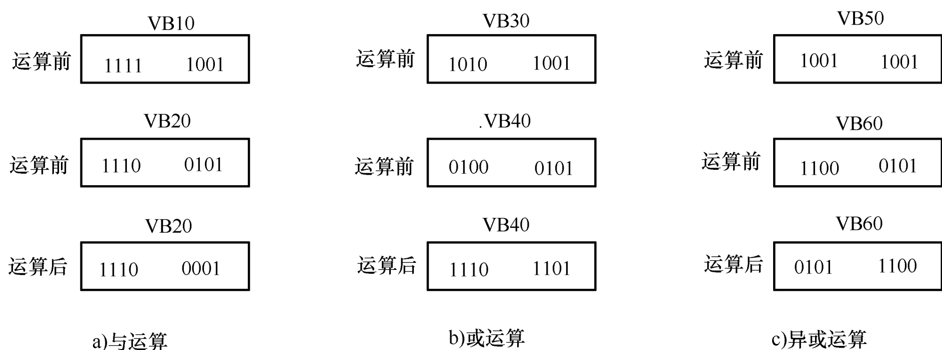


图 4. 9-2　字节与、字节或、字节异或指令的应用

第十节　高速计数器指令

普通计数器是按照顺序扫描的方式进行工作的，在每个扫描周期中，对计数脉冲只能进行一次累加，计数频率一般仅有几十赫兹。然而，在现代自动控制中实现精确定位和测量长度，则需要采用高速计数器来完成对高频率输入信号计数的任务。

一、指令介绍

1. 高速计数器

高速计数器用来累计比 PLC 扫描频率高得多的脉冲输入（S7-200 CPU22X 系列最高计数频率可达到 30kHz)，高速计数器是通过在一定的条件下产生的中断事件完成预定操作的。

（1）高速计数器数量及地址编号　S7-200 CPU22X 中，高速计数器数量及地址编号见表 4. 10-1。

表 4. 10-1　高速计数器数量及地址编号

CPU 类型	CPU221	CPU222	CPU224	CPU226
高速计数器数量	4		6	
高速计数器地址编号	HC0，HC3 ~ HC5		HC0 ~ HC5	

（2）高速计数器的输入端　高速计数器的输入端不像普通输入端那样由用户自由定义，而是由系统指定的。每个高速计数器对它所支持的脉冲输入端、方向控制、复位和启动都有专用的输入点，见表 4. 10-2。

表 4. 10-2　高速计数器的输入点

高速计数器编号	输　入　点	高速计数器编号	输　入　点
HC0	I0. 0、I0. 1、I0. 2	HC3	I0. 1
HC1	I0. 6、I0. 7、I1. 0、I1. 1	HC4	I0. 3、I0. 4、I0. 5
HC2	I1. 2、I1. 3、I1. 4、I1. 5	HC5	I0. 4

表 4.10-2 中用到的输入点，如 I0.0 ~ I0.3，既可以作为普通输入点使用，又可以作为边沿中断输入点使用，还可以在使用高速计数器时作为指定的专用输入点使用，但对于同一个输入点同时只能作为上述其中的一种功能使用。只要使用高速计数器，相应输入点就分配给相应的高速计数器，实现由高速计数器产生的中断。各个高速计数器引起的中断事件见表 4.10-3。

表 4.10-3　高速计数器中断事件

高速计数器编号	当前值等于设定值中断		计数方向改变中断		外部信号复位中断	
	事件号	优先级	事件号	优先级	事件号	优先级
HC0	12	10	27	11	28	12
HC1	13	13	14	14	15	15
HC2	16	16	17	17	4-9	4-9
HC3	32	19	无	无	无	无
HC4	29	20	30	21	31	22
HC5	33	23	无	无	无	无

（3）高速计数器的工作模式　高速计数器有 12 种工作模式，分为以下 4 大类：

模式 0 ~ 模式 2 采用单路脉冲输入的内部方向控制加/减计数；

模式 3 ~ 模式 5 采用单路脉冲输入的外部方向控制加/减计数；

模式 6 ~ 模式 8 采用两路脉冲输入的加/减计数；

模式 9 ~ 模式 11 采用两路脉冲输入的双相正交计数。

每个高速计数器都有多种工作模式，可以通过编程的方法，使用高速计数器定义指令 HDEF 来选择不同的工作模式。

1）高速计数器 0 是一个通用的加/减计数器，共有 8 种工作模式，见表 4.10-4。

表 4.10-4　高速计数器 0 的工作模式

<table>
<tr><th>模式</th><th>描述</th><th>控制位</th><th>I0.0</th><th>I0.1</th><th>I0.2</th></tr>
<tr><td>0</td><td rowspan="2">单路脉冲输入的内部方向控制加/减计数</td><td rowspan="2">SM37.3 = 0，减
SM37.3 = 1，加</td><td rowspan="2">脉冲</td><td rowspan="2"></td><td></td></tr>
<tr><td>1</td><td>复位</td></tr>
<tr><td>3</td><td rowspan="2">单路脉冲输入的外部方向控制加/减计数</td><td rowspan="2">I0.1 = 0，减
I0.1 = 1，加</td><td rowspan="2">脉冲</td><td rowspan="2">方向</td><td></td></tr>
<tr><td>4</td><td>复位</td></tr>
<tr><td>6</td><td rowspan="2">两路脉冲输入的加/减计数
加计数脉冲输入端有输入，加计数
减计数脉冲输入端有输入，减计数</td><td rowspan="2">外部输入控制</td><td rowspan="2">加计数
脉冲</td><td rowspan="2">减计数
脉冲</td><td></td></tr>
<tr><td>7</td><td>复位</td></tr>
<tr><td>9</td><td rowspan="2">两路脉冲输入的双相正交计数
A 超前 B，加计数
B 超前 A，减计数</td><td rowspan="2">外部输入控制</td><td rowspan="2">A 相
脉冲</td><td rowspan="2">B 相
脉冲</td><td></td></tr>
<tr><td>10</td><td>复位</td></tr>
</table>

2）高速计数器 1 共有 12 种工作模式，见表 4.10-5。

表 4.10-5　高速计数器 1 的工作模式

<table>
<tr><th>模　　式</th><th>描　　述</th><th>控　制　位</th><th>I0.6</th><th>I0.7</th><th>I1.0</th><th>I1.1</th></tr>
<tr><td>0</td><td rowspan="3">单路脉冲输入的
内部方向控制加/减计数</td><td rowspan="3">SM47.3 = 0，减
SM47.3 = 1，加</td><td rowspan="3">脉冲</td><td rowspan="3"></td><td></td><td rowspan="2"></td></tr>
<tr><td>1</td><td rowspan="2">复位</td></tr>
<tr><td>2</td><td>启动</td></tr>
<tr><td>3</td><td rowspan="3">单路脉冲输入的
外部方向控制加/减计数</td><td rowspan="3">I0.7 = 0，减
I0.7 = 1，加</td><td rowspan="3">脉冲</td><td rowspan="3">方向</td><td></td><td rowspan="2"></td></tr>
<tr><td>4</td><td rowspan="2">复位</td></tr>
<tr><td>5</td><td>启动</td></tr>
<tr><td>6</td><td rowspan="3">两路脉冲输入的加/减计数
加计数脉冲输入端有输入，加计数
减计数脉冲输入端有输入，减计数</td><td rowspan="3">外部输入控制</td><td rowspan="3">加计数
脉冲</td><td rowspan="3">减计数
脉冲</td><td></td><td rowspan="2"></td></tr>
<tr><td>7</td><td rowspan="2">复位</td></tr>
<tr><td>8</td><td>启动</td></tr>
<tr><td>9</td><td rowspan="3">两路脉冲输入的双相正交计数
A 超前 B，加计数
B 超前 A，减计数</td><td rowspan="3">外部输入控制</td><td rowspan="3">A 相
脉冲</td><td rowspan="3">B 相
脉冲</td><td></td><td rowspan="2"></td></tr>
<tr><td>10</td><td rowspan="2">复位</td></tr>
<tr><td>11</td><td>启动</td></tr>
</table>

3）高速计数器 2 共有 12 种工作模式，见表 4.10-6。

表 4.10-6　高速计数器 2 的工作模式

<table>
<tr><th>模　　式</th><th>描　　述</th><th>控　制　位</th><th>I1.2</th><th>I1.3</th><th>I1.4</th><th>I1.5</th></tr>
<tr><td>0</td><td rowspan="3">单路脉冲输入的
内部方向控制加/减计数</td><td rowspan="3">SM57.3 = 0，减
SM57.3 = 1，加</td><td rowspan="3">脉冲</td><td rowspan="3"></td><td></td><td rowspan="2"></td></tr>
<tr><td>1</td><td rowspan="2">复位</td></tr>
<tr><td>2</td><td>启动</td></tr>
<tr><td>3</td><td rowspan="3">单路脉冲输入的
外部方向控制加/减计数</td><td rowspan="3">I1.3 = 0，减
I1.3 = 1，加</td><td rowspan="3">脉冲</td><td rowspan="3">方向</td><td></td><td rowspan="2"></td></tr>
<tr><td>4</td><td rowspan="2">复位</td></tr>
<tr><td>5</td><td>启动</td></tr>
<tr><td>6</td><td rowspan="3">两路脉冲输入的加/减计数
加计数脉冲输入端有输入，加计数
减计数脉冲输入端有输入，减计数</td><td rowspan="3">外部输入控制</td><td rowspan="3">加计数
脉冲</td><td rowspan="3">减计数
脉冲</td><td></td><td rowspan="2"></td></tr>
<tr><td>7</td><td rowspan="2">复位</td></tr>
<tr><td>8</td><td>启动</td></tr>
<tr><td>9</td><td rowspan="3">两路脉冲输入的双相正交计数
A 超前 B，加计数
B 超前 A，减计数</td><td rowspan="3">外部输入控制</td><td rowspan="3">A 相
脉冲</td><td rowspan="3">B 相
脉冲</td><td></td><td rowspan="2"></td></tr>
<tr><td>10</td><td rowspan="2">复位</td></tr>
<tr><td>11</td><td>启动</td></tr>
</table>

4）高速计数器 3 只有 1 种工作模式，见表 4.10-7。

表 4.10-7　高速计数器 3 的工作模式

模　　式	描　　述	控　制　位	I0.1
0	单路脉冲输入的内部方向控制加/减计数	SM137.3 = 0，减；SM137.3 = 1，加	脉冲

5）高速计数器 4 共有 8 种工作模式，见表 4.10-8。

表 4.10-8 高速计数器 4 的工作模式

模　式	描　述	控 制 位	I0.3	I0.4	I0.5
0	单路脉冲输入的内部方向控制加/减计数	SM147.3 =0，减 SM147.3 =1，加	脉冲		
1					复位
3	单路脉冲输入的外部方向控制加/减计数	I0.4 =0，减 I0.4 =1，加	脉冲	方向	
4					复位
6	两路脉冲输入的加/减计数 加计数脉冲输入端有输入，加计数 减计数脉冲输入端有输入，减计数	外部输入控制	加计数 脉冲	减计数 脉冲	
7					复位
9	两路脉冲输入的双相正交计数 A 超前 B，加计数 B 超前 A，减计数	外部输入控制	A 相 脉冲	B 相 脉冲	
10					复位

6）高速计数器 5 只有 1 种工作模式，见表 4.10-9。

表 4.10-9 高速计数器 5 的工作模式

模　式	描　述	控 制 位	I0.4
0	单路脉冲输入的内部方向控制加/减计数	SM157.3 =0，减；SM157.3 =1，加	脉冲

（4）高速计数器的计数方式

1）单路脉冲输入的内部方向控制加/减计数（模式 0 ~ 模式 2）。即只有一个脉冲输入端，通过高速计数器的控制字节的第 3 位来控制做加计数或者减计数，该位 =1 时为加计数，该位 =0 时为减计数，如图 4.10-1 所示。

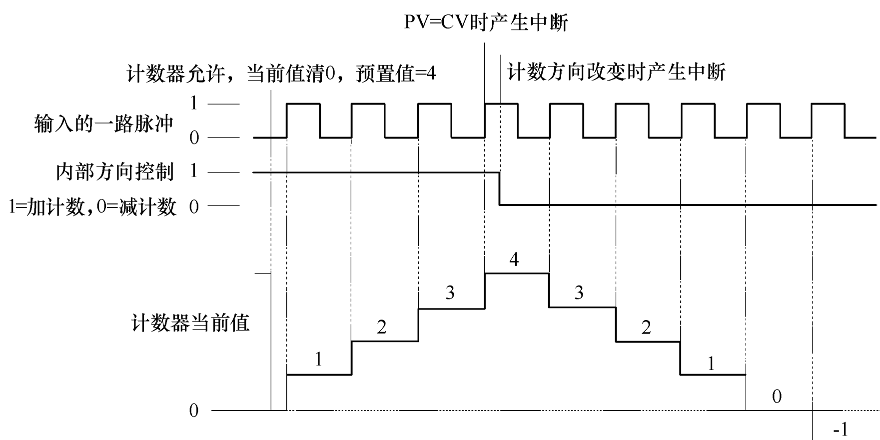


图 4.10-1 单路脉冲输入的内部方向控制加/减计数

2）单路脉冲输入的外部方向控制加/减计数（模式 3 ~ 模式 5）。即有一个脉冲输入端、一个方向控制端，方向输入信号等于 1 时为加计数，等于 0 时为减计数，如图 4.10-2 所示。

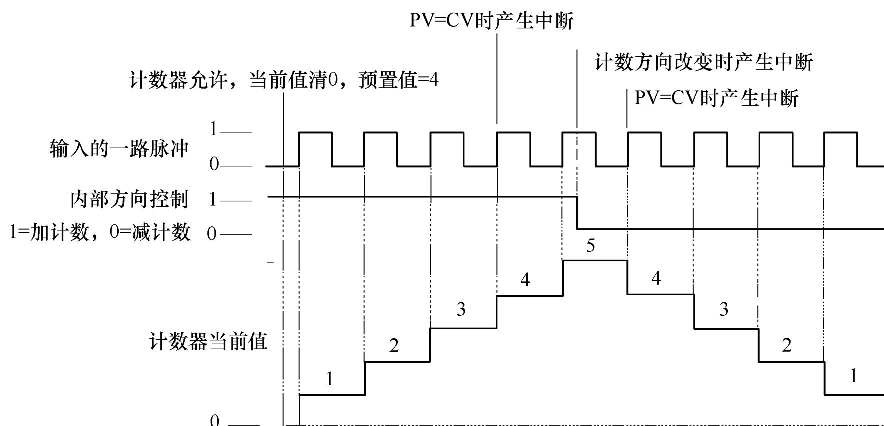


图 4. 10-2　单路脉冲输入的外部方向控制加/减计数

3）两路脉冲输入的加/减计数（模式 6 ~ 模式 8）。即有两个脉冲输入端，一个是加计数脉冲输入端，一个是减计数脉冲输入端，计数值为两个输入端脉冲的代数和，如图 4. 10-3 所示。

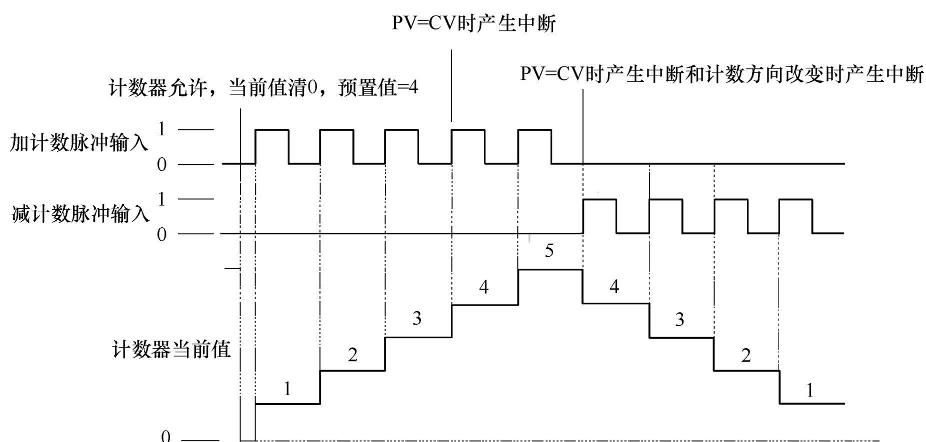


图 4. 10-3　两路脉冲输入的加/减计数

4）两路脉冲输入的双相正交计数（模式 9 ~ 模式 11）。即有两个脉冲输入端，输入的两路脉冲 A 相、B 相，其相位互差 90°（正交），A 相超前 B 相 90°时为加计数，A 相滞后 B 相 90°时为减计数。在这种计数方式下，可选择 1X 模式（单倍频，一个时钟脉冲计一个数）和 4X 模式（四倍频，一个时钟脉冲计四个数），如图 4. 10-4 和图 4. 10-5 所示。

（5）高速计数器的控制字节和状态字节

1）控制字节。定义了计数器和工作模式之后，还要设置高速计数器的有关控制字节。每个高速计数器均有一个控制字节，它决定了计数器的计数允许或禁用、方向控制（仅限模式 0、1 和 2）或对所有其他模式的初始化计数方向、装入当前值和预置值。控制字节见表 4. 10-10。

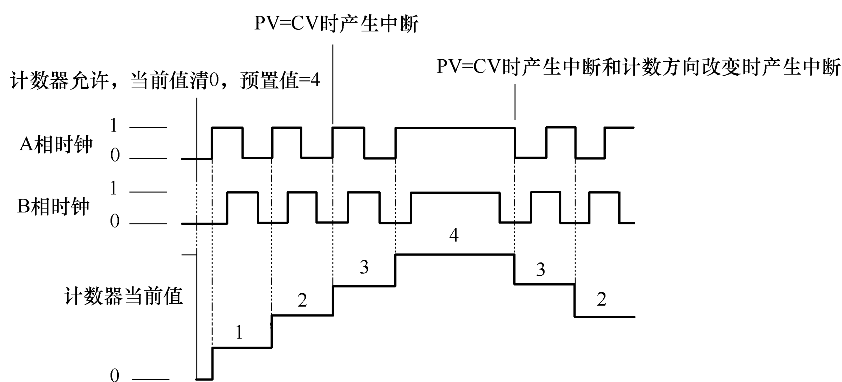


图 4.10-4　两路脉冲输入的双相正交计数（1X 模式）

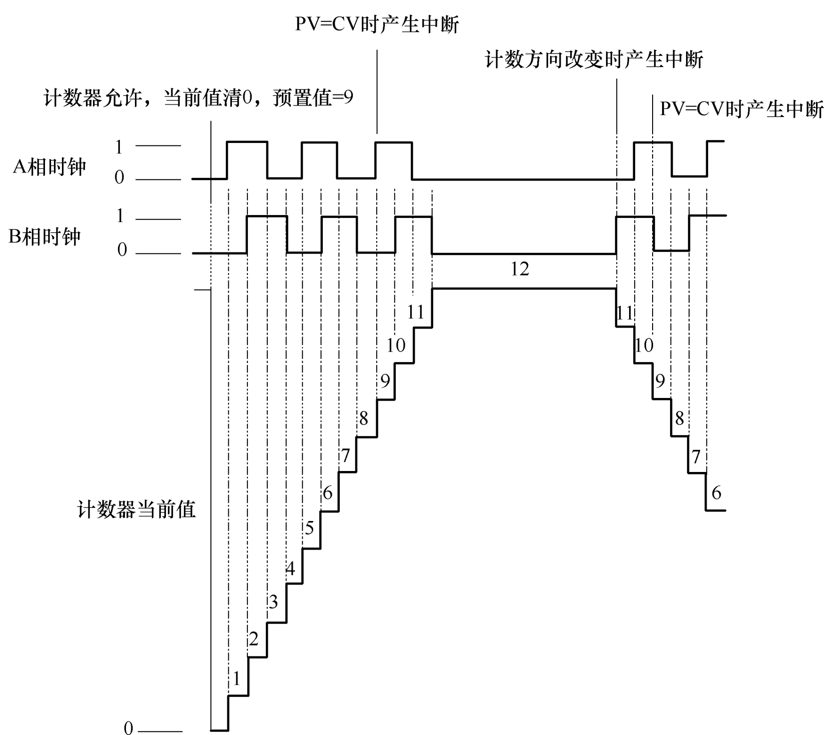


图 4.10-5　两路脉冲输入的双相正交计数（4X 模式）

表 4.10-10　高速计数器的控制字节

HC0	HC1	HC2	HC3	HC4	HC5	说　　明
SM37.0	SM47.0	SM57.0		SM147.0		复位有效电平控制： 0 = 复位信号高电平有效；1 = 低电平有效
	SM47.1	SM57.1				启动有效电平控制： 0 = 启动信号高电平有效；1 = 低电平有效

（续）

HC0	HC1	HC2	HC3	HC4	HC5	说 明
SM37.2	SM47.2	SM57.2		SM147.2		正交计数器计数速率选择： 0 = 4X 计数速率；1 = 1X 计数速率
SM37.3	SM47.3	SM57.3	SM137.3	SM147.3	SM157.3	计数方向控制位： 0 = 减计数，1 = 加计数
SM37.4	SM47.4	SM57.4	SM137.4	SM147.4	SM157.4	向 HC 写入计数方向： 0 = 无更新，1 = 更新计数方向
SM37.5	SM47.5	SM57.5	SM137.5	SM147.5	SM157.5	向 HC 写入新预置值： 0 = 无更新，1 = 更新预置值
SM37.6	SM47.6	SM57.6	SM137.6	SM147.6	SM157.6	向 HC 写入新当前值： 0 = 无更新，1 = 更新当前值
SM37.7	SM47.7	SM57.7	SM137.7	SM147.7	SM157.7	HC 允许： 0 = 禁用 HC；1 = 启用 HC

2）状态字节。每个高速计数器都有一个状态字节，状态位表示当前计数方向以及当前值是否大于或等于预置值。状态字节的 0 ~ 4 位不用。监控高速计数器状态的目的是使外部事件产生中断，以完成重要的操作。状态字节见表 4.10-11。

表 4.10-11 高速计数器的状态字节

HC0	HC1	HC2	HC3	HC4	HC5	说 明
SM36.5	SM46.5	SM56.5	SM136.5	SM146.5	SM156.5	当前计数方向状态位： 0 = 减计数；1 = 加计数
SM36.6	SM46.6	SM56.6	SM136.6	SM146.6	SM156.6	当前值等于预置值状态位： 0 = 不相等；1 = 等于
SM36.7	SM46.7	SM56.7	SM136.7	SM146.7	SM156.7	当前值大于预置状态位： 0 = 小于或等于；1 = 大于

（6）高速计数器寻址 如果要指定高速计数器的地址，访问高速计数器的当前值（CV），就要使用存储器类型 HC 和计数器编号，即 HC0、HC1、HC2、HC3、HC4、HC5。高速计数器的当前值（HC0 ~ HC5）为只读数据，其数据长度为双字（32 位）。由于 HC 数据类型为只读，故不能使用 HC 数据类型将一个新当前计数写入高速计数器。

2. 高速计数器指令介绍

高速计数器指令包括高速计数器定义（HDEF）指令和高速计数器执行（HSC）指令。

（1）高速计数器定义（HDEF）指令 HDEF 指令指定高速计数器的工作模式。每个高速计数器只能用一条高速计数器定义指令。HDEF 指令的梯形图和语句表见

表 4. 10-12。

（2）高速计数器执行（HSC）指令　HSC 指令根据高速计数器控制位的状态和 HDEF 指令指定的工作模式，控制高速计数器执行高速计数工作，参数 N 指定高速计数器的号码。HSC 指令的梯形图和语句表见表 4. 10-12。

表 4. 10-12　高速计数器指令的梯形图和语句表

指令名称	梯形图	指　令	功　能	数据类型及操作数
高速计数器定义指令 HDEF	HDEF EN ENO ????-HSC ????-MODE	HDEF HSC，MODE	使能输入有效时，为指定的高速计数器分配一种工作模式，并且只能定义一次	数据类型：字节型 操作数：HSC，高速计数器编号，为 0 ~ 5 的常数；MODE，工作模式，为 0 ~ 11 的常数
高速计数器执行指令 HSC	HSC EN ENO ????-N	HSC N	使能输入有效时，根据高速计数器特殊存储器位的状态，并按照 HDEF 指令指定的工作模式，设置高速计数器并控制其工作	数据类型：字型 N：高速计数器编号，为 0 ~ 5 的常数

（3）高速计数器指令的使用　每个高速计数器都有一个 32 位当前值和一个 32 位预置值，当前值和预设值均为带符号整数值。当前值和预置值占用的特殊内部标志位存储区见表 4. 10-13。

表 4. 10-13　当前值和预置值占用的特殊内部标志位存储区

要装入的数值	HC0	HC1	HC2	HC3	HC4	HC5
新的当前值	SMD38	SMD48	SMD58	SMD138	SMD148	SMD158
新的预置值	SMD42	SMD52	SMD62	SMD142	SMD152	SMD162

要设置高速计数器的新当前值和新预置值，必须先设置控制字节，令其第 5 位和第 6 位为 1，允许更新预置值和当前值，新当前值和新预置值写入特殊内部标志位存储区，然后执行 HSC 指令，从而将新数值传送到高速计数器。

执行 HDEF 指令前，必须将高速计数器控制字节的位设置成需要的状态，否则将采用默认设置。默认设置：复位和启动输入均为高电平有效，正交计数速率选择 4X 模式。执行 HDEF 指令后，就不能再改变计数器的设置，除非 CPU 进入停止模式。

执行 HSC 指令时，CPU 检查控制字节和有关的当前值和预置值。

（4）高速计数器的初始化步骤　高速计数器的初始化即对高速计数器设置控制字节、执行高速计数器定义指令（选择工作模式）、设定当前值和预置值、设置中断和执行高速计数器执行指令等。因为 HDEF 指令进入 RUN 模式后只能执行一次，为了减少程序运行时间优化程序结构，一般以子程序的形式进行高速计数器的初始化。当然，也可以用主程序

中的程序段来实现。高速计数器在运行之前，必须要执行一次初始化子程序或初始化程序段。

高速计数器的初始化步骤如下：

1）用首次扫描时接通一个扫描周期的特殊内部存储器 SM0.1 去调用一个子程序，完成初始化操作。

2）在初始化子程序中，根据希望的控制设置控制字节（SMB37、SMB47、SMB57、SMB137、SMB147、SMB157）。如选择高速计数器 1，设置 SMB47 = 16#FC，则为允许计数、写入新当前值、写入新预置值、更新计数方向为加计数、正交计数设为单倍频（1X）模式、复位和启动均设置为高电平有效。

3）执行 HDEF 指令，设置 HC 的编号（0 ~ 5），设置工作模式（0 ~ 11）。如 HC 的编号设置为 1，工作模式输入设置为 11，则为既有复位又有启动的正交计数工作模式。

4）将新的当前值写入 32 位当前值寄存器（SMD38、SMD48、SMD58、SMD138、SMD148、SMD158）。如写入 0，则清除当前值，用指令“MOVD 0，SMD48”实现。

5）将新的预置值写入 32 位预置值寄存器（SMD42、SMD52、SMD62、SMD142、SMD152、SMD162）。如执行指令“MOVD 1000，SMD52”，则设置预置值为 1000。若写入预置值为 16#00，则高速计数器处于不工作状态。

6）为了捕捉当前值等于预置值的事件，将条件 CV = PV 中断事件（如选择 HC1，则为事件 13）与一个中断程序相联系。

7）为了捕捉计数方向的改变，将计数方向改变的中断事件（如选择 HC1，则为事件 14）与一个中断程序相联系。

8）为了捕捉外部信号复位，将外部信号复位中断事件（如选择 HC1，则为事件 15）与一个中断程序相联系。

9）执行全局中断允许指令（ENI），允许 HC 中断。

10）执行 HSC 指令，激活高速计数器。

11）结束子程序。

二、指令练习

某设备采用位置编码器作为检测元件，需要高速计数器进行位置值的计数，其要求如下：计数信号为 A、B 两相相位差 90°的脉冲输入；使用外部计数器复位与启动信号，高电平有效；编码器每转的脉冲数为 2500，在 PLC 内部进行 4 倍频，计数开始值为“0”，当转动 1 转后，需要清除计数值进行重新计数。

1. 主程序（见图 4.10-6）

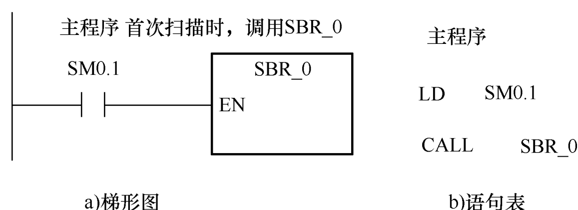


图 4.10-6　主程序

2. 初始化的子程序（见图 4. 10-7）

3. 中断程序（见图 4. 10-8）

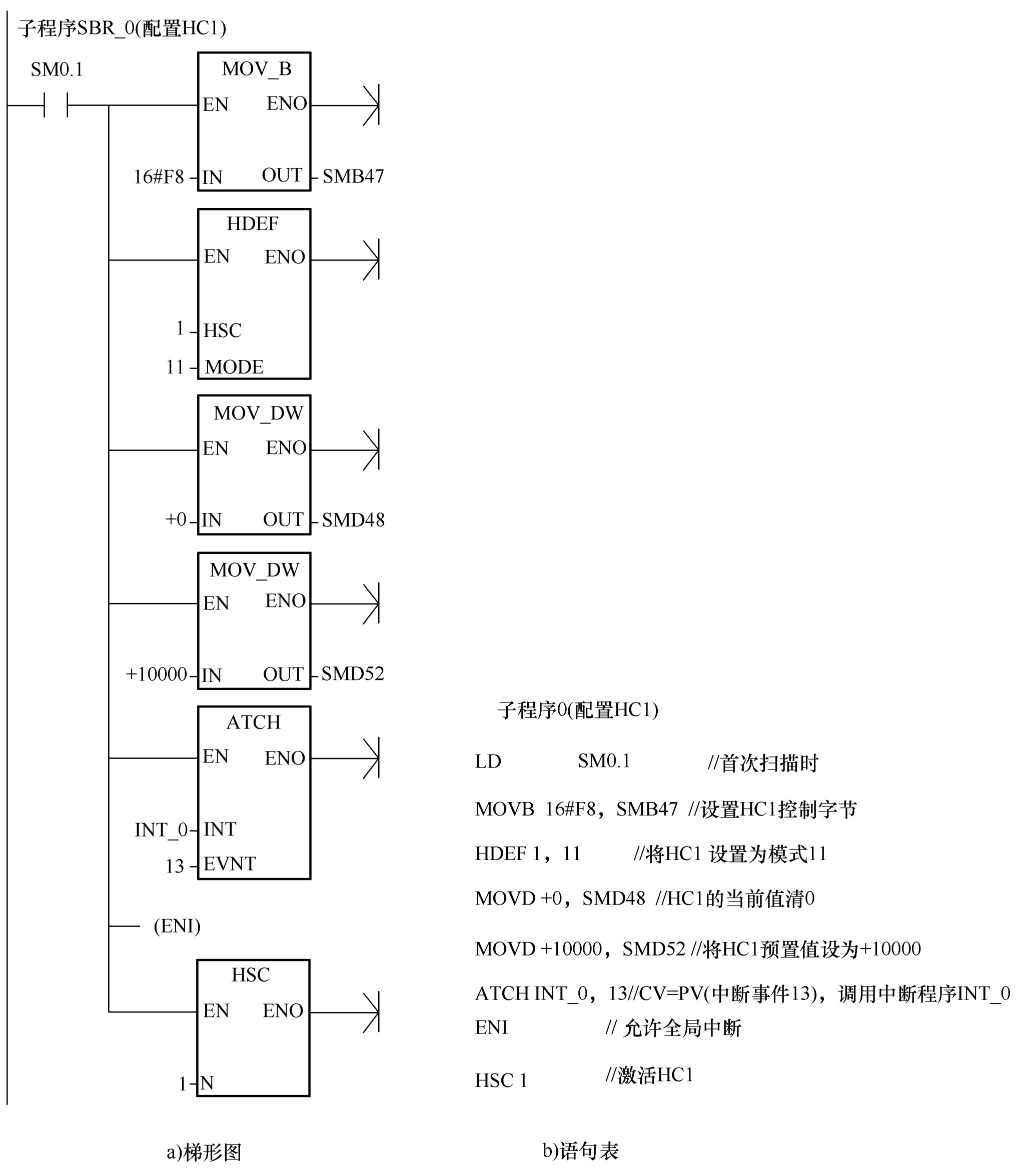


图 4. 10-7 初始化的子程序

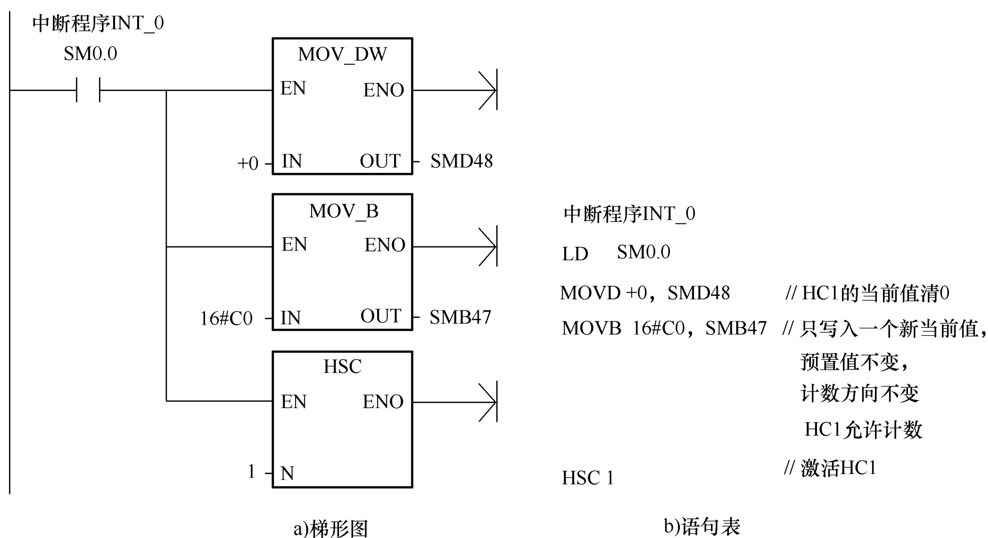


图 4. 10-8　中断程序

第十一节　高速脉冲输出指令

高速脉冲输出功能是指在 PLC 中输出高速脉冲，用来驱动负载的精确控制，在运动控制中广泛应用。使用高速脉冲输出功能时，PLC 主机应选用晶体管输出型，以满足高速输出的频率要求。

一、指令介绍

1. 高速脉冲输出形式

S7-200 晶体管输出型的 PLC 有高速脉冲串输出（PTO）和脉冲宽度调制出（PWM）两种方式。

（1）PTO（Pulse Train Output，脉冲串输出）　PTO 能输出频率可调的、占空比为 50% 的一串脉冲如图 4. 11-1 所示（用户可以控制脉冲周期和脉冲数），多用于带有位置控制功能的步进驱动器或伺服驱动器。将输出脉冲的个数作为位置给定值输入，以实现定位控制功能；通过改变定位脉冲的输出频率，可以改变运动的速度。

（2）PWM（Pulse Width Modulation，脉冲宽度调制）　PWM 能输出占空比可调的一串脉冲如图 4. 11-2 所示（用户可以控制脉冲周期和脉冲宽度），用于直接驱动调速系统或运动控制系统的输出级，控制逆变主回路。

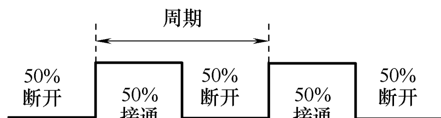


图 4. 11-1　PTO 输出形式

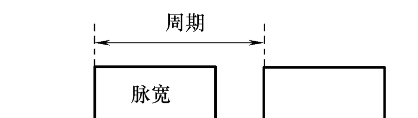


图 4. 11-2　PWM 输出形式

2. 高速脉冲输出端子的确定

S7-200 PLC 的高速脉冲输出端子是不能随意选择的，必须按照系统指定的输出点 Q0.0 和 Q0.1 来选择。当 Q0.0 或 Q0.1 设置为 PTO 或 PWM 功能时，Q0.0 或 Q0.1 输出点就不能当作普通的数字量输出使用，其输出波形不受输出过程映像寄存器的状态、输出强制或立即输出指令的影响。当不使用 PTO/PWM 发生器功能时，输出点 Q0.0、Q0.1 使用通用功能，输出由输出过程映像寄存器控制，输出过程映像寄存器决定输出信号波形的起始和结束状态，即决定脉冲输出波形从高电平或低电平开始和结束，使输出波形有短暂的不连续，为了减小这种不连续的有害影响，应注意以下两点。

1）可在起用 PTO 或 PWM 操作之前，将用于 Q0.0 和 Q0.1 的输出过程映像寄存器设为 0。

2）PTO/PWM 输出必须至少有 10% 的额定负载，才能完成从关闭至打开以及从打开至关闭的顺利转换，即提供陡直的上升沿和下降沿。

3. 脉冲输出 PLS 指令

（1）PLS 指令的功能　PLS 指令使 PLC 输出端产生高速脉冲，用来驱动负载实现精确控制，如对步进电动机的控制。PLS 指令的梯形图及语句表见表 4.11-1。

表 4.11-1　PLS 指令的梯形图和语句表

指令名称	梯形图	语句表	操作数及数据类型
PLS 指令	PLS EN　ENO Q0.X	PLS　Q	Q：常量（0 或 1） 数据类型：字

PLS 指令功能描述：使能有效时，检查用于脉冲输出（Q0.0 或 Q0.1）的特殊存储器位（SM），然后执行特殊存储器位定义的脉冲操作，从 Q0.0 或 Q0.1 输出高速脉冲。

（2）用于脉冲输出（Q0.0 或 Q0.1）的特殊标志寄存器　每个 PTO/PWM 发生器都对应一定数量的特殊寄存器，这些寄存器包括控制字节寄存器、状态字节寄存器和参数数值寄存器，见表 4.11-2，用它们来控制高速脉冲的输出形式和输出状态及参数值。

表 4.11-2　相关寄存器

Q0.0 的寄存器	Q0.1 的寄存器	名称及功能描述
SMB66	SMB76	状态字节，在 PTO 方式下，跟踪脉冲串的输出状态
SMB67	SMB77	控制字节，控制 PTO/PWM 脉冲输出的基本功能
SMW68	SMW78	PTO/PWM 的周期值，字型，范围为 2～65 535，16 位无符号数
SMW70	SMW80	PWM 的脉宽值，字型，范围为 0～65 535，16 位无符号数
SMD72	SMD82	PTO 的脉冲数，双字型，范围为 1～4 294 967 295，32 位无符号数
SMB166	SMB176	多段管线 PTO 进行中的段的编号，8 位无符号数
SMW168	SMW178	多段管线 PTO 包络表起始字节的地址

说明：表 4.11-2 中 SMB67 控制 PTO/PWM 的 Q0.0，SMB77 控制 PTO/PWM 的 Q0.1，

SMW68/SMW78、SMW70/SMW80、SMD72/SMD82 分别存放周期值、脉冲宽度值、脉冲数值。在多段脉冲串操作中，执行 PLS 指令前应在 SMB166/SMB176 中填入管线的总段数，在 SMWl68/SMWl78 中装入包络表的起始偏移地址并填好包络表的值。状态字节用于监视 PTO 发生器的工作。

PLS 指令从 PTO/PWM 对应的寄存器中读取数据，使程序按寄存器中的值控制 PTO/PWM 发生器。因此执行 PLS 指令前，必须设置好这些寄存器。寄存器各位的功能见表 4.11-3。表 4.11-4 作为一个快速参考，用其中的数值作为 PTO/PWM 控制字节的值来实现需要的操作。

表 4.11-3　PTO/PWM 控制寄存器

	Q0.0	Q0.1	说　明
状态字节	SM66.4	SM76.4	PTO 包络由于增量计算错误异常终止，0：无错；1：异常终止
	SM66.5	SM76.5	PTO 包络由于用户命令异常终止，0：无错；1：异常终止
	SM66.6	SM76.6	PTO 管线溢出，0：无溢出；1：溢出
	SM66.7	SM76.7	PTO 空闲（用来指示脉冲序列输出结束），0：运行中；1：PTO 空闲
控制字节	SM67.0	SM77.0	PTO/PWM 刷新周期值，0：不刷新；1：刷新
	SM67.1	SM77.1	PWM 刷新脉冲宽度值，0：不刷新；1：刷新
	SM67.2	SM77.2	PTO 刷新脉冲计数值，0：不刷新；1：刷新
	SM67.3	SM77.3	PTO/PWM 时基选择，0：1μs；1：1ms
	SM67.4	SM77.4	PWM 更新方法，0：异步更新；1：同步更新
	SM67.5	SM77.5	PTO 操作，0：单段操作；1：多段操作
	SM67.6	SM77.6	PTO/PWM 模式选择，0：选择 PTO　1：选择 PWM
	SM67.7	SM77.7	PTO/PWM 允许，0：禁止；1：允许
其他寄存器	SMW68	SMW78	PTO/PWM 周期时间值（范围：2～65 535）
	SMW70	SMW80	PWM 脉冲宽度值（范围：0～65 535）
	SMD72	SMD82	PTO 脉冲计数值（范围：1～4 294 967 295）
	SMB166	SMB176	操作中的段数（仅用在多段 PTO 操作中）
	SMW168	SMW178	包络表的起始位置，用从 V0 开始的字节偏移量表示（仅用在多段 PTO 操作中）

表 4.11-4　PTO/PWM 控制字节参考

控制字节（十六进制）	执行 PLS 指令的结果							
	启　用	模式选择	PTO 段操作	PWM 更新方法	时　基	脉冲数	脉冲宽度	周　期
16#81	是	PTO	单段		1μs/周期			装载
16#84	是	PTO	单段		1μs/周期	装载		
16#85	是	PTO	单段		1μs/周期	装载		装载
16#89	是	PTO	单段		1ms/周期			装载
16#8C	是	PTO	单段		1ms/周期	装载		

（续）

控制字节（十六进制）	执行 PLS 指令的结果							
	启　用	模式选择	PTO 段操作	PWM 更新方法	时　基	脉冲数	脉冲宽度	周　期
16#8D	是	PTO	单段		1ms/周期	装载		装载
16#A0	是	PTO	多段		1μs/周期			
16#A8	是	PTO	多段		1ms/周期			
16#D1	是	PWM		同步	1μs/周期			装载
16#D2	是	PWM		同步	1μs/周期		装载	
16#D3	是	PWM		同步	1μs/周期		装载	装载
16#D9	是	PWM		同步	1ms/周期			装载
16#DA	是	PWM		同步	1ms/周期		装载	
16#DB	是	PWM		同步	1ms/周期		装载	装载

4. PTO 的应用

PTO 按照给定的脉冲个数和周期输出一串方波（占空比 50%）。PTO 可以产生单段脉冲串或者多段脉冲串，来指定脉冲数和周期（以微秒或毫秒为增加量）。

（1）周期和脉冲数　周期的范围是 50～65 535μs 或 2～65 535ms，为 16 位无符号数，时基有 μs 和 ms 两种，通过控制字节的第 3 位选择。如果设定的周期是奇数，会引起占空比的一些失真。脉冲数的范围是 1～4 294 967 295。

如果周期时间小于最小值，就把周期默认为最小值。如果指定脉冲数为 0，就把脉冲数默认为 1 个脉冲。

状态字节中的 PTO 空闲位（SM66.7 或 SM76.7）为 1 时，则指示脉冲串输出完成，可根据脉冲串输出的完成调用中断程序。

（2）单段管线模式　PTO 功能可输出多个脉冲串，允许脉冲串的排队，形成管线。当激活的脉冲串输出完成后，立即开始输出新的脉冲串，保证了脉冲串顺序输出的连续性。

PTO 发生器有单段管线和多段管线（也称单段流水线和多段流水线）两种模式。

单段管线是指管线中每次只能存储一个脉冲串的控制参数。一旦启动了 PTO 起始段，就必须立即为下一个脉冲串更新控制寄存器，并再次执行 PLS 指令。第二个脉冲串的属性在管线一直保持到第一个脉冲串发送完成。第一个脉冲串发送完成，紧接着就输出第二个脉冲串，重复上述过程可输出多个脉冲串。输出多个脉冲串时，如果采用单段管线模式，编程比较复杂。

单段管线模式中的各段脉冲串可以采用不同的时间基准，但有可能造成脉冲串之间的不平稳过渡。但是，如果时间基准发生了变化或者在利用 PLS 指令捕捉到新脉冲之前，启动的脉冲串已经完成，那么脉冲串之间则可以做到平滑转换。

当管线满时，如果试图装入另一个脉冲串的控制参数，状态寄存器中的 PTO 溢出位（SM66.6 或 SM76.6）将置位。在检测到溢出后，必须手动清除这个位，以便恢复检测功能。当 PLC 进入 RUN 方式时，这个位初始化为 0。

（3）单段 PTO 的编程步骤　通常，用一个子程序为单段操作的脉冲输出配置和初始化 PTO，从主程序调用初始化子程序。使用首次扫描内存位（SM0.1）将脉冲输出初始化为 0，

并调用子程序，执行初始化操作。使用子程序调用时，随后的扫描不再调用该子程序，这样会降低扫描时间执行，程序结构更为合理。

从主程序建立初始化子程序调用后，用以下步骤建立控制逻辑，用于在初始化子程序中配置脉冲输出 Q0.0 或 Q0.1。

1）设置控制字节。将值 16#85（选择微秒增加）或值 16#8D（选择 ms 增加）写入 SMB67 或 SMB77，两个值均可启用 PTO/PWM 功能、选择 PTO 操作、设置脉冲数和周期值以及选择时基（μs 或 ms）。

2）在 SMW68 或 SMW78 中写入一个周期的字值。

3）在 SMD72 或 SMD82 中写入脉冲计数的双字值。

4）（选项）如果希望在脉冲串输出完成后立即执行相关功能，可以将脉冲串完成事件（中断事件 19）附加于中断子程序，为中断编程，使用 ATCH 指令并执行全局中断启用指令 ENI。

5）执行 PLS 指令，使 S7-200 为 PTO 发生器编程。

6）退出子程序。

对于单段 PTO 操作，如果要修改 PTO 的周期、脉冲数，可以在子程序或中断程序中执行：

1）根据要修改的内容，设置相应的控制字节值。

2）在 SMW68 或 SMW78 中写入新周期的一个字值。

3）在 SMD72 或 SMD82 中写入新脉冲计数的一个双字值。

4）执行 PLS 指令，使 S7-200 为 PTO 发生器编程。用更新脉冲计数和脉冲时间信号波形输出开始之前，CPU 必须完成所有进行中的 PTO。

5）退出子程序或中断程序。

（4）多段管线模式

1）包络表。多段管线中，CPU 在变量存储区（V）建立一个包络表。包络表中存储各个脉冲串的控制参数。多段管线用 PLS 指令启动。多段管线作业中，CPU 自动从包络表中按顺序读出每个脉冲串的控制参数，并实施脉冲串输出。当执行 PLS 指令时，包络表内容不可改变。

包络表由包络段数和各段参数构成，包络表的格式见表 4.11-5。

表 4.11-5　多段 PTO 操作的包络表格式

从包络表开始的字节偏移	包 络 段 数	说明（注：周期增量值 Δ 为整数微秒或毫秒）
0		段数（1～255）；数 0 产生一个非致命性错误，将不产生 PTO 输出
1	#1	初始周期（2～65 535 时间基准单位）
3		每个脉冲的周期增量（有符号数，－32 768～32 767 时间基准单位）
5		脉冲数（1～4 294 967 295）
9	#2	初始周期（2～65 535 时间基准单位）
11		每个脉冲的周期增量（有符号数，－32 768～32 767 时间基准单位）
13		脉冲数（1～4 294 967 295）
……	……	……

包络表每段的长度都是 8 个字节，由周期值（16bit）、周期增量值（16bit）和脉冲计数值（32bit）组成。包络表中的参数表征了脉冲串的特性。在包络表中周期增量可以选择 μs

或 ms，但在同一个包络表中的所有周期值必须使用同一个时间基准。

多段 PTO 操作的特点是编程简单，能够按照每个脉冲的个数自动增减周期。周期增量区的值为正值，则增加周期；周期增量区的值为负值，则减少周期；周期增量区的值为 0，则周期不变。除周期增量为 0 外，每个输出脉冲的周期值都发生着变化。

如果在输出若干个脉冲后指定的周期增量值导致非法周期值，会产生溢出错误，则 SM66.6 或 SM76.6 被置为 1，同时停止 PTO 功能，PLC 的输出变为通用功能。另外，状态字节中的增量计算错误位（SM66.4 或 SM76.4）被置为 1。

如果要人为地终止一个正进行中的 PTO 包络，只需要把状态字节中的用户终止位（SM66.5 或 SM76.5）置为 1。

当 PTO 包络执行时，当前启动的段的编号保存在 SMB166（或 SMB176）。

2）计算包络表中的数值。PTO 发生器的多段管线功能在实际应用中非常有用。例如，步进电动机的运动控制要求如图 4.11-3 所示。从 A 点到 B 点为起动加速过程，从 B 到 C 为恒速运行过程，从 C 到 D 为减速停止过程。根据控制要求列出 PTO 包络表。

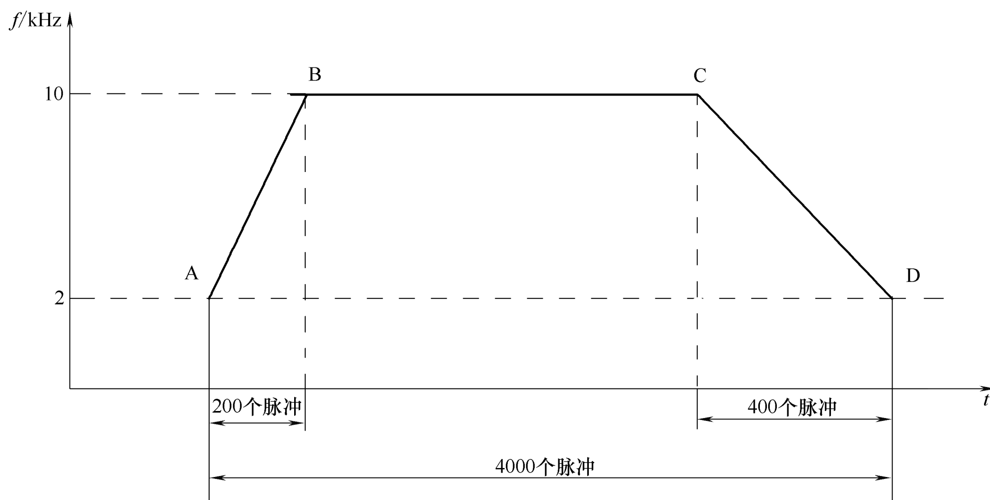


图 4.11-3　步进电动机的运动控制要求

步进电动机的运动控制分为 3 段（起动加速、恒速运行、减速停止），共需要 4 000 个脉冲，即属于多段管线，需建立 3 段脉冲的包络表。起动和终止脉冲频率为 2kHz，最大脉冲频率为 10kHz。由于包络表中的值是用周期表示的，而不是用频率，需要把给定的频率值转换成周期值。起始和终止周期为 500μs，相应于最大频率的周期为 100μs。段 1：加速运行，应在约 200 个脉冲时达到最大脉冲频率（10kHz）；段 2：恒速运行，约 4000 − 200 − 400 = 3400 个脉冲；段 3：减速运行，应在约 400 个脉冲时完成。

PTO 发生器用来调整某一特定段每次脉冲周期的周期增量值 Δ 为

$$周期增量值\ \Delta = (ECT - ICT)/Q$$

其中，ECT 为该段的结束周期，ICT 为该段的初始周期，Q 为该段中的脉冲数量。

利用上式计算出段 1 的周期增量值 Δ 为 −2，段 2 的周期增量值 Δ 为 0，段 3 的周期增量值 Δ 为 1。假设包络表存放在从 VB200 开始的 V 存储器区，则包络表见表 4.11-6。该包络

表给出了产生所要求信号波形的值，该包络表的值可以在用户程序中用指令放在 V 存储器中。

表 4.11-6　包络表

V 变量存储器地址	参数值	说　明	
VB200	3	段数	
VW201	500	初始周期	段 1
VW203	-2	周期增量 Δ	
VD205	200	脉冲数	
VW209	100	初始周期	段 2
VW211	0	周期增量 Δ	
VD213	3 400	脉冲数	
VW217	100	初始周期	段 3
VW219	1	周期增量 Δ	
VD221	400	脉冲数	

（5）多段 PTO 的编程步骤　用一个子程序实现 PTO 初始化，首次扫描（SM0.1）时从主程序调用初始化子程序，执行初始化操作。

初始化操作步骤如下：

1）首次扫描（SM0.1）时将输出 Q0.0 或 Q0.1 复位（置 0），并调用完成初始化操作的子程序。

2）在初始化子程序中，根据控制要求设置控制字节并写入 SMB67 或 SMB77 特殊存储器。如写入 16#A0（选择 μs 递增）或 16#A8（选择 ms 递增），两个数值表示允许 PTO 功能、选择 PTO 操作、选择多段操作以及选择时基（μs 或 ms）。

3）将包络表的首地址（16 位）写入 SMW168（或 SMW178）。

4）在变量存储器 V 中，写入包络表的各参数值。一定要在包络表的起始字节中写入段数。在变量存储器 V 中建立包络表的过程也可以在一个子程序中完成，在此只需调用设置包络表的子程序。

5）设置中断事件并全局开中断。如果想在 PTO 完成后，立即执行相关功能，则必须设置中断，将脉冲串完成事件（中断事件号 19）连接一中断程序。

6）执行 PLS 指令，使 S7-200 为 PTO 发生器编程，高速脉冲串由 Q0.0 或 Q0.1 输出。

7）退出子程序。

5. PWM 的应用

PWM 用来输出占空比可调的高速脉冲，用户可以控制脉冲的周期和脉冲宽度来完成特定的控制任务。

（1）周期和脉冲宽度　周期和脉宽时基为 μs 或 ms，均为 16 位无符号数。周期的范围为 50 ~ 65 535μs 或 2 ~ 65 535ms。若周期小于两个时基，则系统默认为两个时基。脉宽范围为 0 ~ 65 535μs 或 0 ~ 65 535ms。若脉宽大于等于周期，占空比等于 100%，输出连续接通；若脉宽等于 0，占空比为 0%，则输出断开。

（2）更新方式　改变 PWM 波形的方法有两种：同步更新和异步更新。

同步更新：不需改变时基时，可以用同步更新。执行同步更新时，波形的变化发生在周

期的边缘，形成平滑转换。

异步更新：需要改变 PWM 的时基时，则应使用异步更新。异步更新使高速脉冲输出功能被瞬时禁用，与 PWM 波形不同步，这样会引起被控设备的振动。鉴于此原因，建议尽量使用同步更新，选择一个适合于所有周期时间的时间基准。

（3）PWM 的编程步骤　通常，用一个子程序为单段操作的脉冲输出配置和初始化 PWM，从主程序调用初始化子程序。使用首次扫描内存位（SM0.1）将脉冲输出初始化为 0，并调用子程序，执行初始化操作。使用子程序调用时，随后的扫描不再调用该子程序，这样会降低扫描时间执行，程序结构更为合理。

从主程序建立初始化子程序调用后，用以下步骤建立控制逻辑，用于在初始化子程序中配置脉冲输出 Q0.0 或 Q0.1。

1）用首次扫描位（SM0.1）使输出位复位为 0，并调用初始化子程序。这样可减少扫描时间，程序结构更合理。

2）在初始化子程序中设置控制字节。如将 16#D3（时基为 μs）或 16#DB（时基为 ms）写入 SMB67 或 SMB77，控制功能为允许 PTO/PWM 功能、选择 PWM 操作、设置更新脉冲宽度和周期数值以及选择时基（μs 或 ms）。

3）在 SMW68 或 SMW78 中写入一个字长的周期值。

4）在 SMW70 或 SMW80 中写入一个字长的脉宽值。

5）执行 PLS 指令，使 S7-200 为 PWM 发生器编程，并由 Q0.0 或 Q0.1 输出。

6）可为下一输出脉冲预设控制字节。在 SMB67 或 SMB77 中写入 16#D2（μs）或 16#DA（ms），控制字节中将禁止改变周期值，允许改变脉宽值。以后只要装入一个新的脉宽值，不用改变控制字节，直接执行 PLS 指令就可改变脉宽值。

7）退出子程序。

二、指令练习

练习 1：单段管线 PTO 的编程练习

有一启动按钮接于 I0.1，停止按钮接于 I0.0。要求当按下启动按钮时，Q0.1 输出 PTO 高速脉冲，脉冲的周期为 100μs，个数为 50 000 个。若输出脉冲过程中按下停止按钮，则脉冲输出立即停止。编写单段管线 PTO 脉冲输出程序，如图 4.11-4 所示。

练习 2：多段管线 PTO 的编程练习

根据多段管线 PTO 初始化和操作步骤，应该按照以下步骤进行编程。

1）编程前首先选择高速脉冲输出端子为 Q0.0，并确定 PTO 为 3 段管线，设置控制字节 SMB67 为 16#A0（表示允许 PTO 功能、选择 PTO 操作、选择多段操作、选择时基为 μs，不允许更新周期和脉冲数）。

2）建立 3 段的包络表（见表 4.11-6），并将包络表的首地址装入 SMW168。

3）PTO 完成调用中断程序，使 Q0.6 接通。

4）PTO 完成的中断事件号为 19。用中断调用指令 ATCH 将中断事件 19 与中断程序 INT_0连接，并全局开中断。

5）执行 PLS 指令，退出子程序。

本练习题的主程序、初始化子程序、中断程序如图 4.11-5 所示。

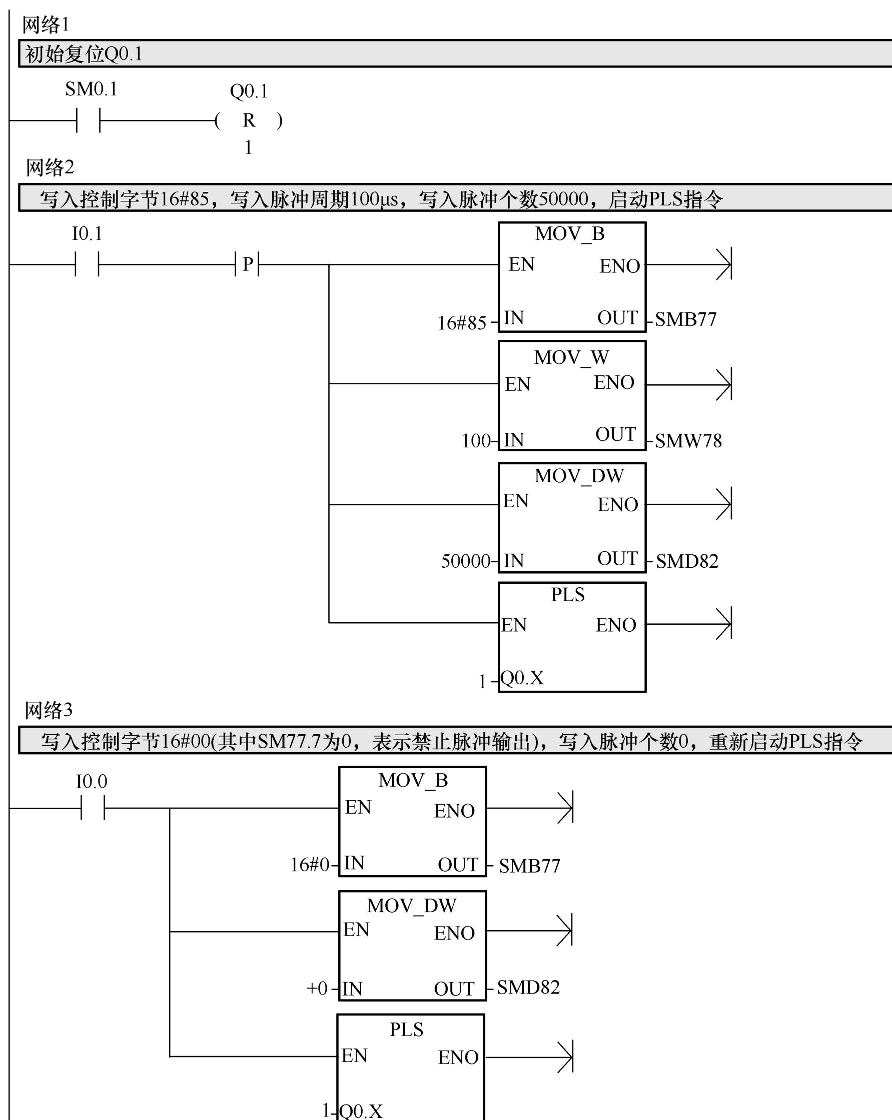


图 4. 11-4　单段管线 PTO 脉冲输出程序

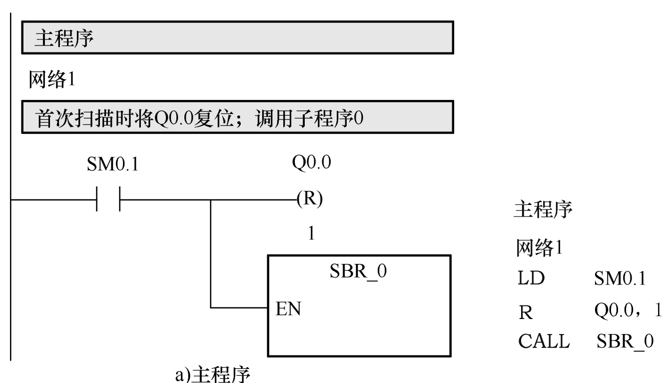
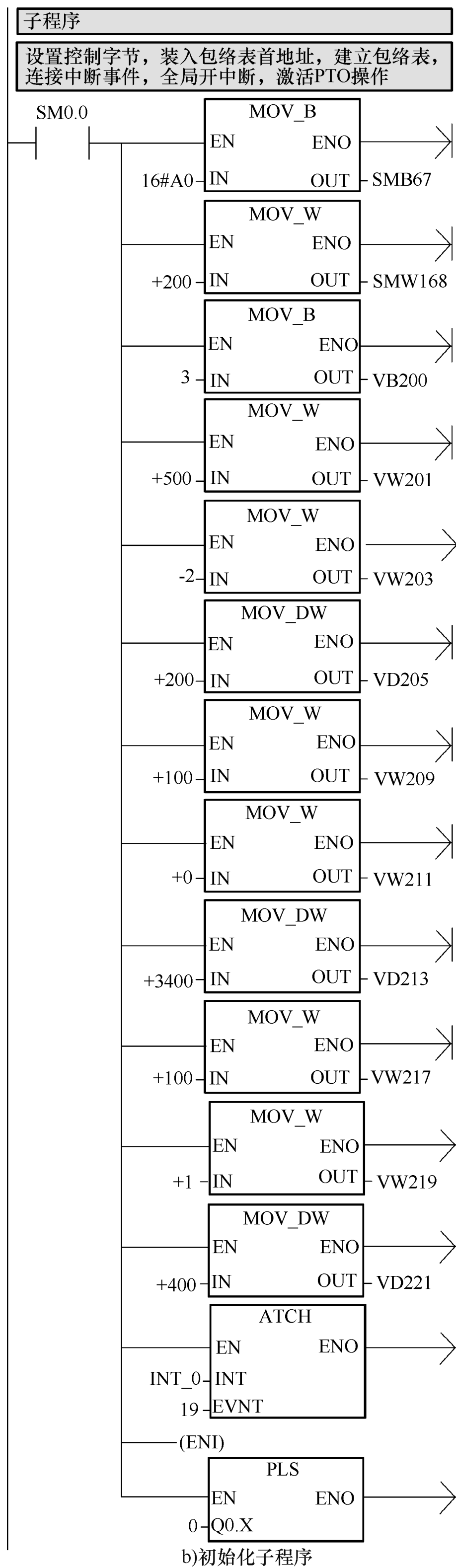
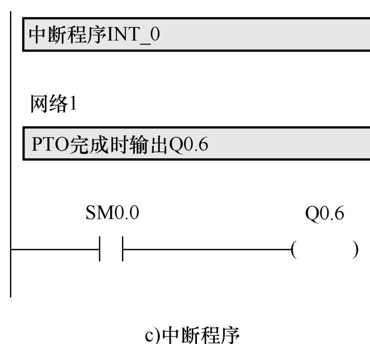


图 4. 11-5　使用多段管线 PTO 的编程实现步进电动机的控制



b)初始化子程序

图 4.11-5　使用多段管线 PTO 的编程实现步进电动机的控制（续）



c)中断程序

图 4. 11-5　使用多段管线 PTO 的编程实现步进电动机的控制（续）

练习 3：PWM 的编程练习

从 PLC 的 Q0. 0 输出高速脉冲，其脉宽的初始值为 0. 1s，周期固定为 1s，其脉宽每周期递增 0. 1s，当脉宽达到设定的 0. 9s 时，脉宽改为每周期递减 0. 1s，直到脉宽减为 0 为止，重新开始循环执行。

因为每个周期都有脉冲宽度变化的操作，要使得波形特性的变化发生在周期边沿，形成波形的平滑转换，一般的做法是将 PWM 输出反馈到 1 个中断输入点，如 I0. 0，即把 Q0. 0 接到 I0. 0（见图 4. 11-6）。当需要改变脉冲宽度时产生中断，在下 1 个 I0. 0 的上升沿，脉冲宽度的改变将与 PWM 的新周期同步发生。

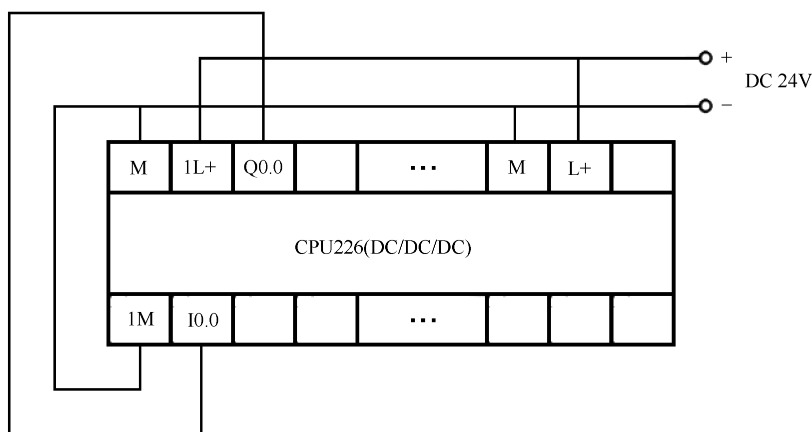


图 4. 11-6　同步更新脉冲宽度的 PWM 输出控制的 PLC 外部接线图

编写两个中断程序，一个中断程序实现脉宽递增，一个中断程序实现脉宽递减，并设置标志位 M0. 0，在初始化操作时使其置位，执行脉宽递增中断程序，当脉宽达到 0. 9s 时，使其复位，执行脉宽递减中断程序。

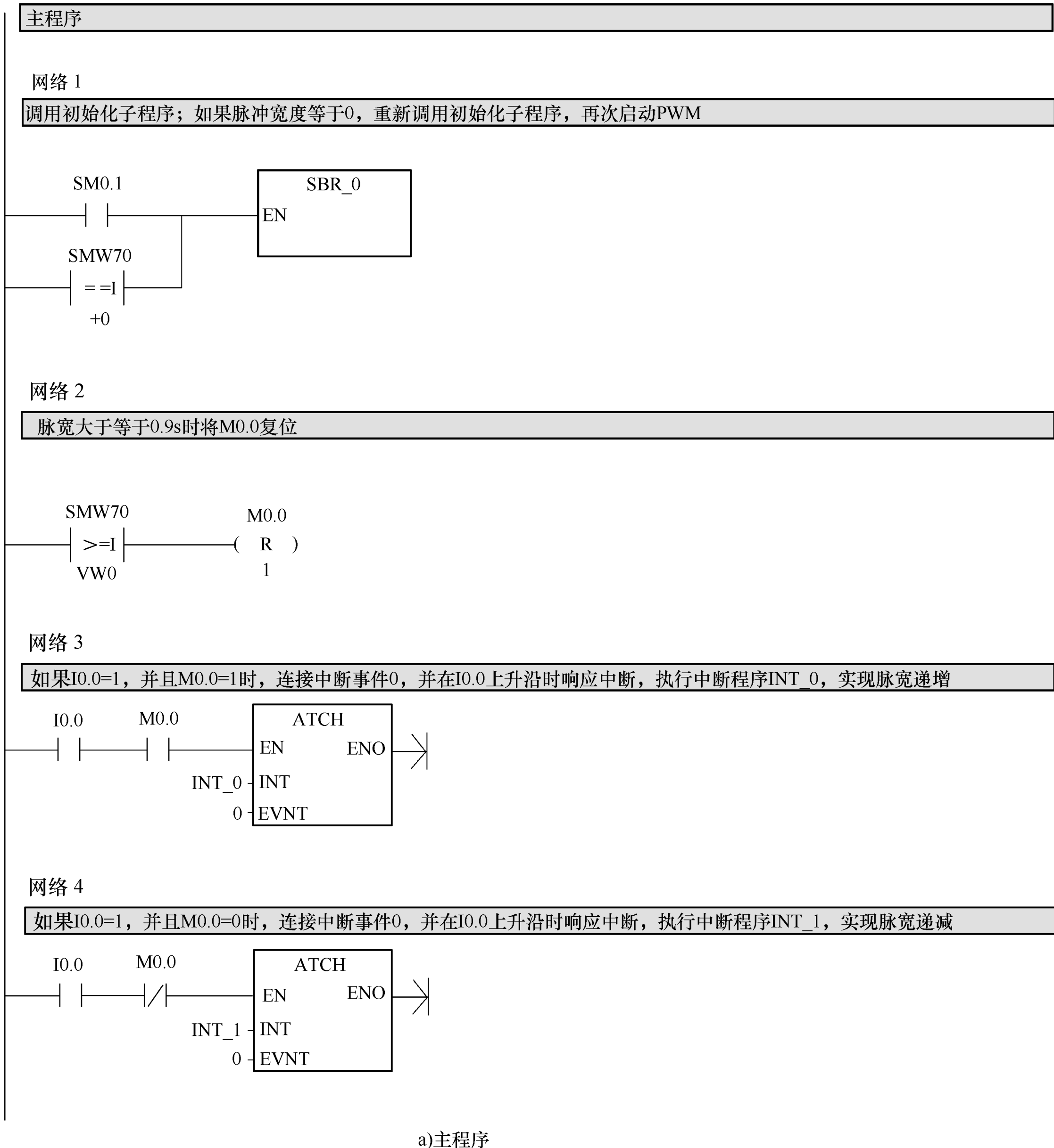
在子程序中完成 PWM 的初始化操作，选用输出端 Q0. 0，控制字节 SMB67 设定为 16#CB（2#11001011，允许 PWM 输出、Q0. 0 为 PWM 方式、异步更新、时基为 ms、允许更新脉宽、允许更新周期）。

在 PWM 输出形式下的典型操作是当前周期为固定值时改变脉冲宽度。在初始化子程序中设置 PWM 的初始值时，一定要先将控制字节的最低位 SM67. 0 或 SM77. 0 设置为 1，初始

化时刷新脉冲周期时间；将 SM67.1 或 SM77.1 设置为 1，初始化时刷新脉冲宽度。即使控制要求的是周期不变化的脉冲输出，但如果在初始化中没有刷新周期时间和脉冲宽度，则系统将自动定为默认值 0，这样就会导致没有正常激活高速脉冲输出。

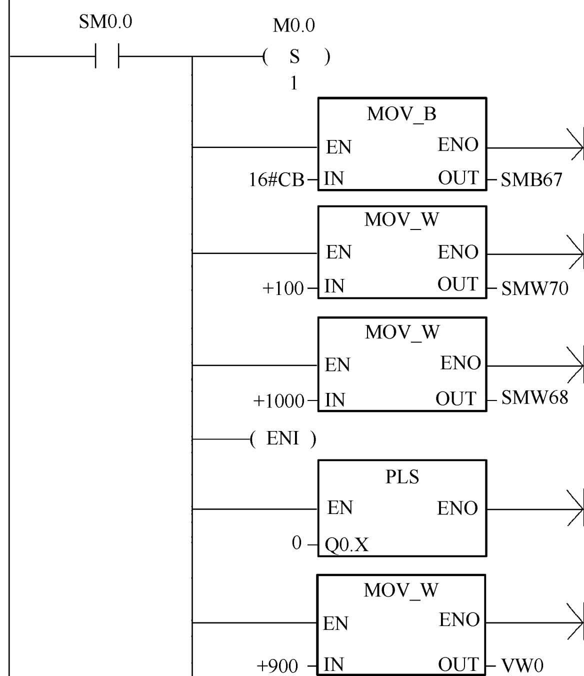
在改变脉冲宽度时，按照是否需要改变时间基准，有同步更新与异步更新两种方式。本练习中，如果采用同步更新方式，当脉冲脉宽减为 0 时，不能调用初始化子程序，高速 PWM 脉冲输出不能重新开始循环。因此，当高速脉冲宽度发生变化时有减至为 0 的情况下，实际工程应用中不能正常循环输出脉冲时，应考虑将控制字节中相关位设置为异步更新方式。

程序如图 4.11-7 所示。

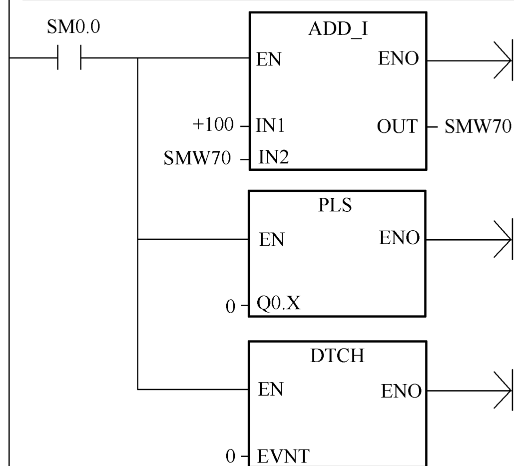


a)主程序

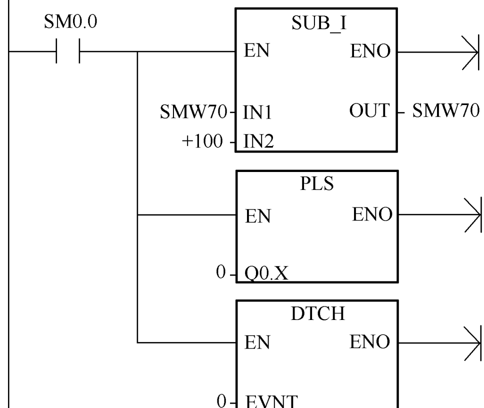
图 4.11-7　PWM 的编程梯形图



b)初始化子程序



c)中断程序INT_0



d) 中断程序INT_1

图 4. 11-7 PWM 的编程梯形图（续）

第十二节　功能指令综合应用实例

实例 1　彩灯的闪烁控制

1. 控制要求

利用四则运算指令中的乘法指令控制 8 位彩灯的双数灯逐盏点亮。当接通 PLC 电源后，输出彩灯按照 2、4、6、8 位进行逐次点亮。

2. 操作步骤

1）根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例只需要 8 个输出点，见表 4.12-1。

表 4.12-1　I/O 分配表

输出继电器	功　能
QB0	控制 8 个彩灯

2）根据控制要求，设计并绘制 PLC 系统接线原理图，如图 4.12-1 所示。

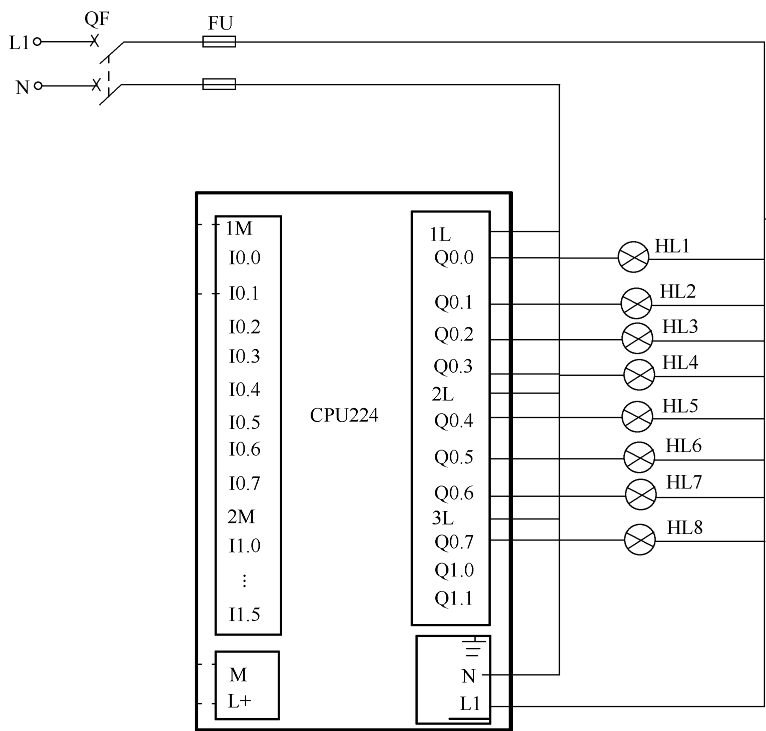


图 4.12-1　PLC 接线原理图

3）程序设计如图 4.12-2 所示。

4）程序输入与调试。把图 4.12-2 所示的程序传入 PLC 中，将 QB0 输出端子分别和指示灯相连接。接通电源，每隔 0.5s 双数灯 2、4、6、8 就逐次点亮。

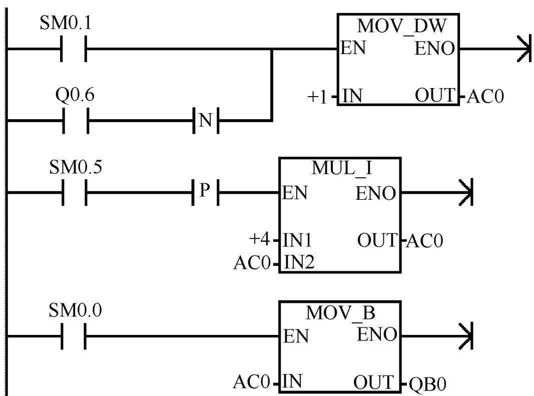


图 4. 12-2　梯形图

实例 2　四路抢答器控制

1. 控制要求

设有四路抢答器，共有四位选手，一位主持人，主持人控制一个开始按钮和一个复位按钮。当主持人按下开始按钮后，选手开始抢答，抢答正常的选手指示灯亮并用数码管显示序号，其他按钮不起作用。如果主持人未按下开始按钮，就有选手抢答，则认为犯规，犯规指示灯闪烁，同时选手序号在数码管上显示。当主持人按下开始按钮后，在 10s 内无人抢答，则系统超时指示灯亮，此题作废选手不能再抢答。所有出现的情况，只有主持人按下系统复位按钮后，才能重新开始。

2. 操作步骤

1）根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 6 个输入点，13 个输出点，见表 4. 12-2。

表 4. 12-2　PLC 的 I/O 配置

输入设备		输入继电器	输出设备		输出继电器
代　号	功　能		代　号	功　能	
SB1	开始按钮	I0. 0	HL1	无人抢答灯	Q0. 0
SB2	复位按钮	I0. 1	HL2	犯规指示灯	Q0. 1
SB3	1 组按钮	I0. 2	HL3	1 组指示灯	Q0. 2
SB4	2 组按钮	I0. 3	HL4	2 组指示灯	Q0. 3
SB5	3 组按钮	I0. 4	HL5	3 组指示灯	Q0. 4
SB6	4 组按钮	I0. 5	HL6	4 组指示灯	Q0. 5
				七段码显示 a 段	Q1. 0
				七段码显示 b 段	Q1. 1
				七段码显示 c 段	Q1. 2
				七段码显示 d 段	Q1. 3
				七段码显示 e 段	Q1. 4
				七段码显示 f 段	Q1. 5
				七段码显示 g 段	Q1. 6

2）根据控制要求，设计并绘制 PLC 系统接线原理图，如图 4.12-3 所示。

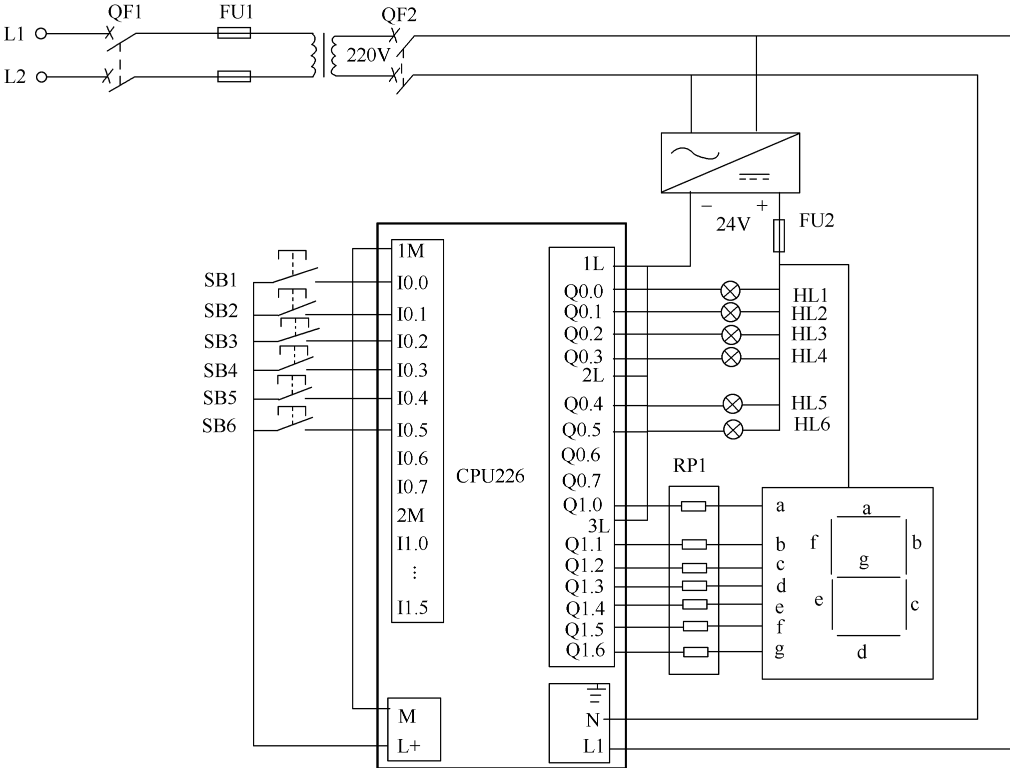


图 4.12-3　PLC 系统接线原理图

3）带数码管显示的四路抢答器的 PLC 梯形图程序如图 4.12-4 所示。

4）程序输入与调试。熟练地操作编程软件，能正确将编制的程序输入 PLC；按照被控设备的要求进行调试、修改，达到设计要求。

实例 3　花样喷泉控制

1. 控制要求

在很多公园、广场、旅游景点或建筑前面会有一些喷泉，并配有五颜六色的灯光和音乐，为群众休闲提供了良好的环境。下面介绍一种常见的喷泉的 PLC 控制方式，如图 4.12-5 所示。

图 4.12-5 中，从里向外分别为 1#喷头（中间）、2#喷头、3#喷头、4#喷头。

按下起动按钮，喷泉控制装置开始工作；按下停止按钮，喷泉控制装置停止工作。喷泉的工作方式有两种，通过方式选择开关来选择。

方式一：开始工作时，1#喷头喷水 3s，接着 2#喷头喷水 3s，然后 3#喷头喷水 3s，最后 4#喷头喷水 20s；重复上述过程，直至按下停止按钮。

方式二：开始工作时，1#、3#喷头喷水 5s，接着 2#、4#喷头喷水 5s，停 2s，如此交替运行 60s，然后 4 组喷头全喷水 20s；重复上述过程，直至按下停止按钮。

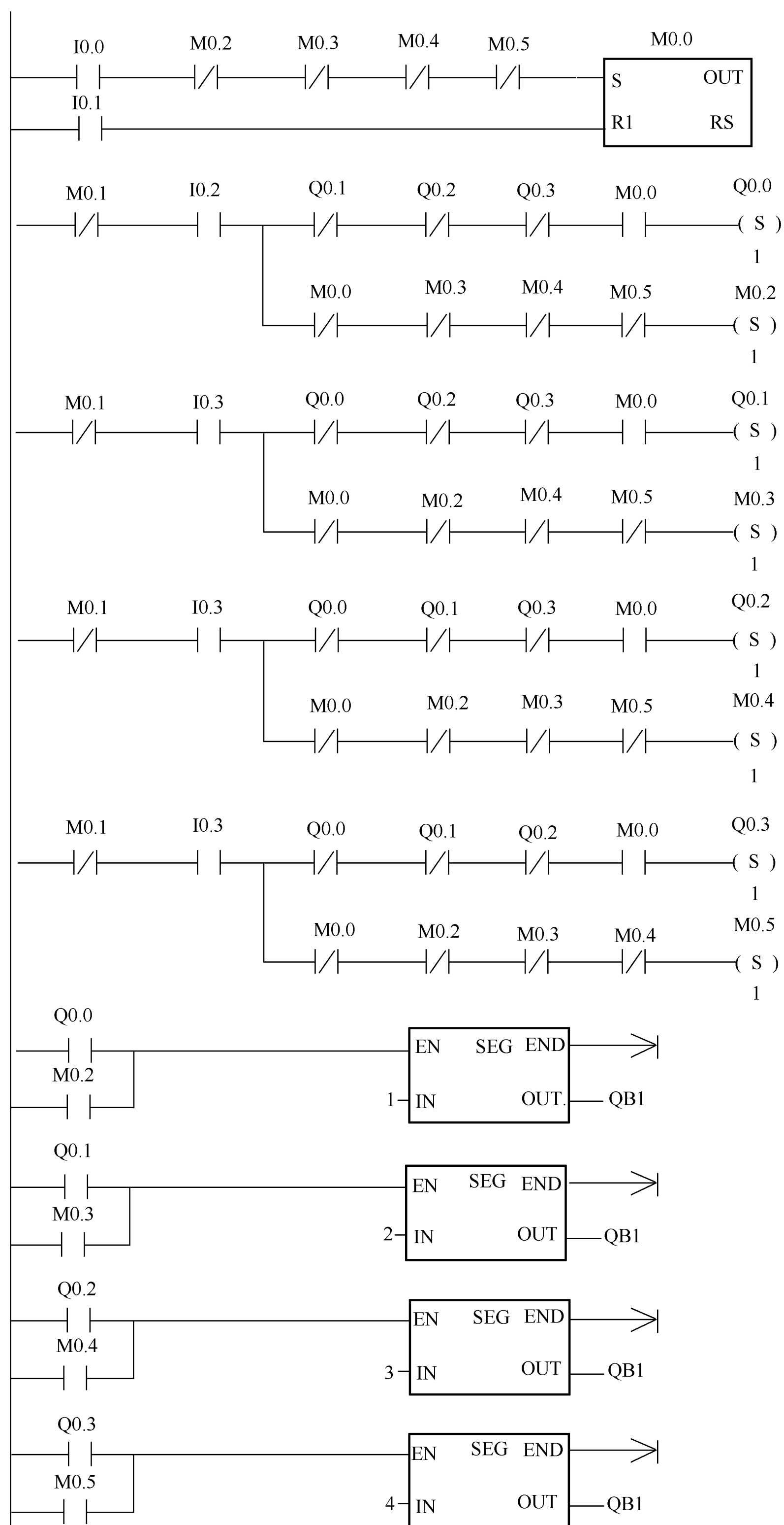


图 4. 12-4　四路抢答器的 PLC 梯形图程序

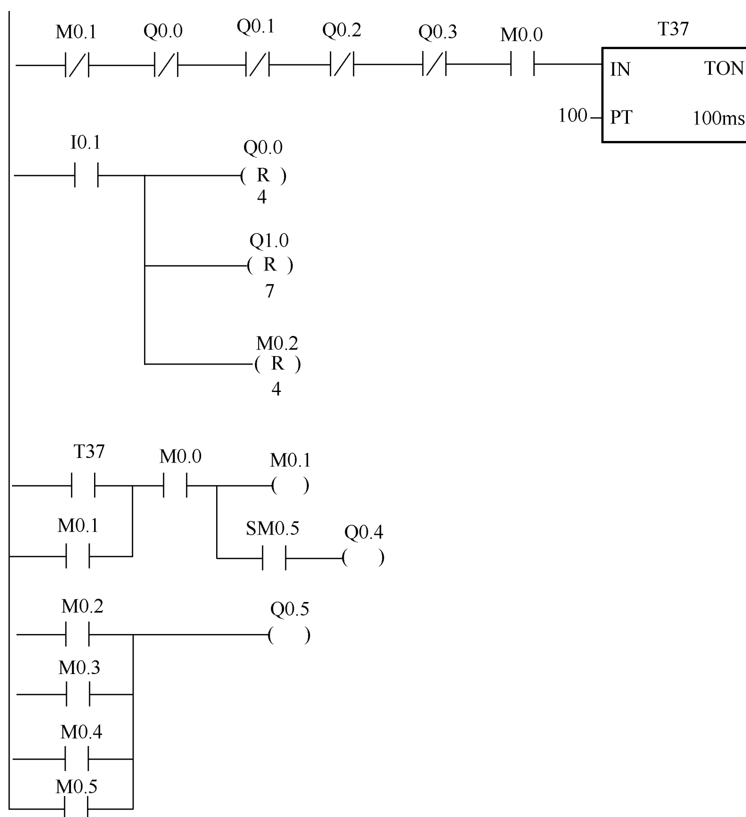


图 4. 12-4　四路抢答器的 PLC 梯形图程序（续）

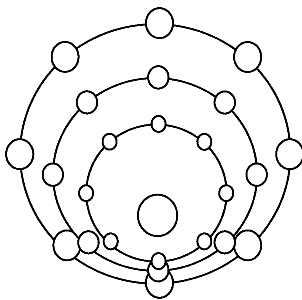


图 4. 12-5　喷泉示意图

2. 操作步骤

1）根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 3 个输入点，4 个输出点，见表 4. 12-3。

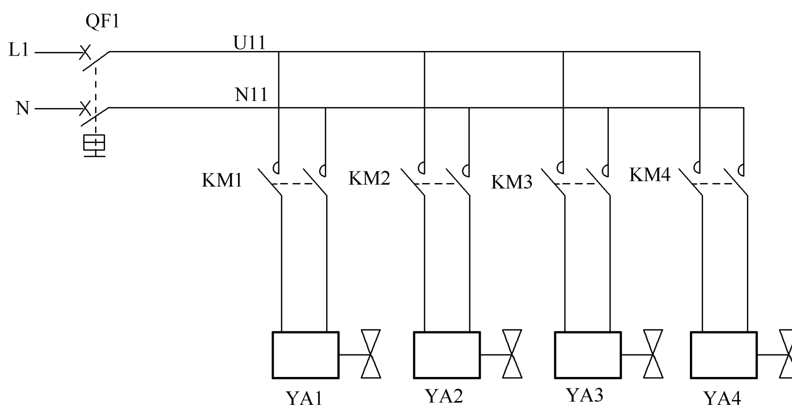
2）根据控制要求，设计并绘制 PLC 系统接线原理图，如图 4. 12-6 所示。

3）梯形图程序如图 4. 12-7 所示。

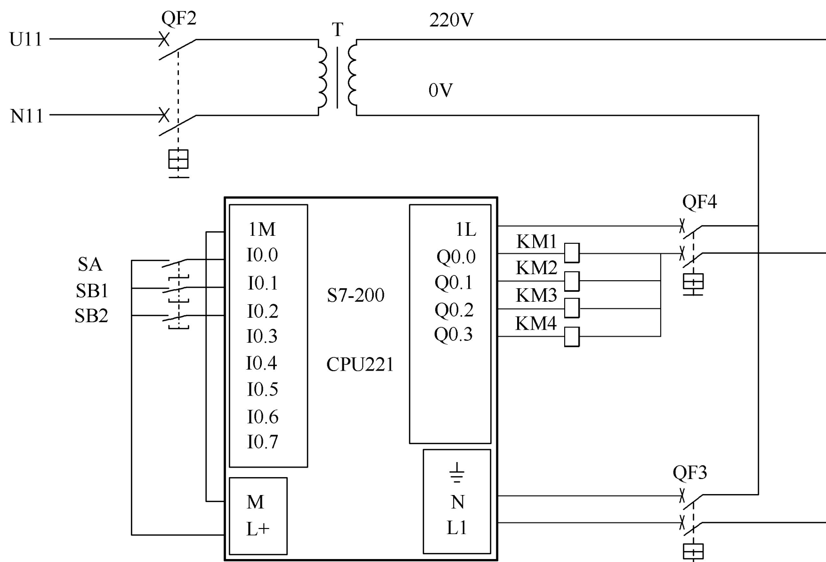
4）程序输入与调试。熟练地操作编程软件，能正确将编制的程序输入 PLC；按照被控设备的要求进行调试、修改，达到设计要求。

表 4.12-3　PLC 的 I/O 配置

输入设备			输出设备		
代　号	功　能	输入继电器	代　号	功　能	输出继电器
SA	方式选择开关	I0.0	KM1	控制 1#喷头	Q0.0
SB1	停止	I0.1	KM2	控制 2#喷头	Q0.1
SB2	起动	I0.2	KM3	控制 3#喷头	Q0.2
			KM4	控制 4#喷头	Q0.3



a) PLC系统接线原理图1



b) PLC系统接线原理图2

图 4.12-6　PLC 接线原理图

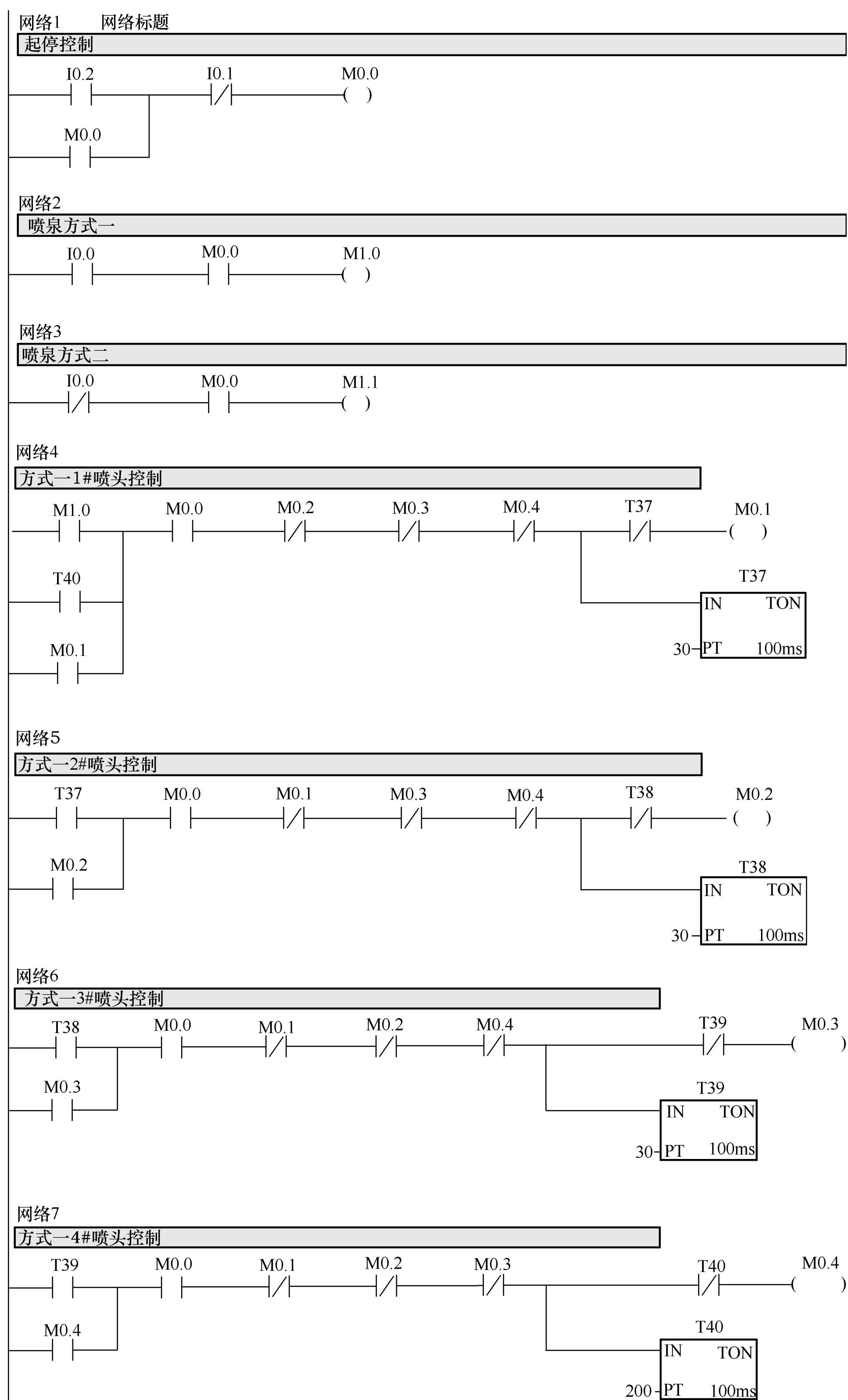


图 4. 12-7 喷泉的 PLC 梯形图程序

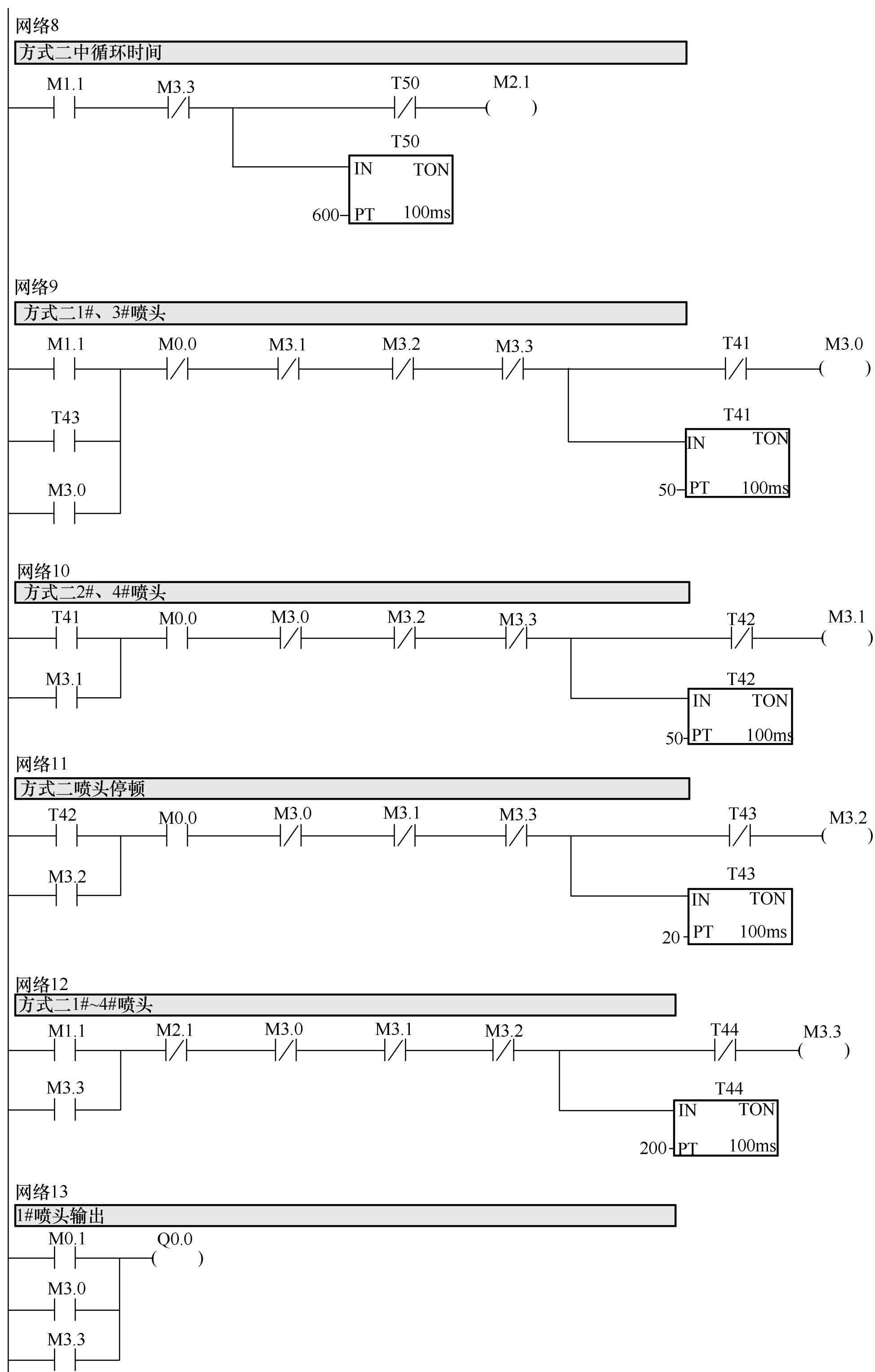


图 4. 12-7 喷泉的 PLC 梯形图程序（续）

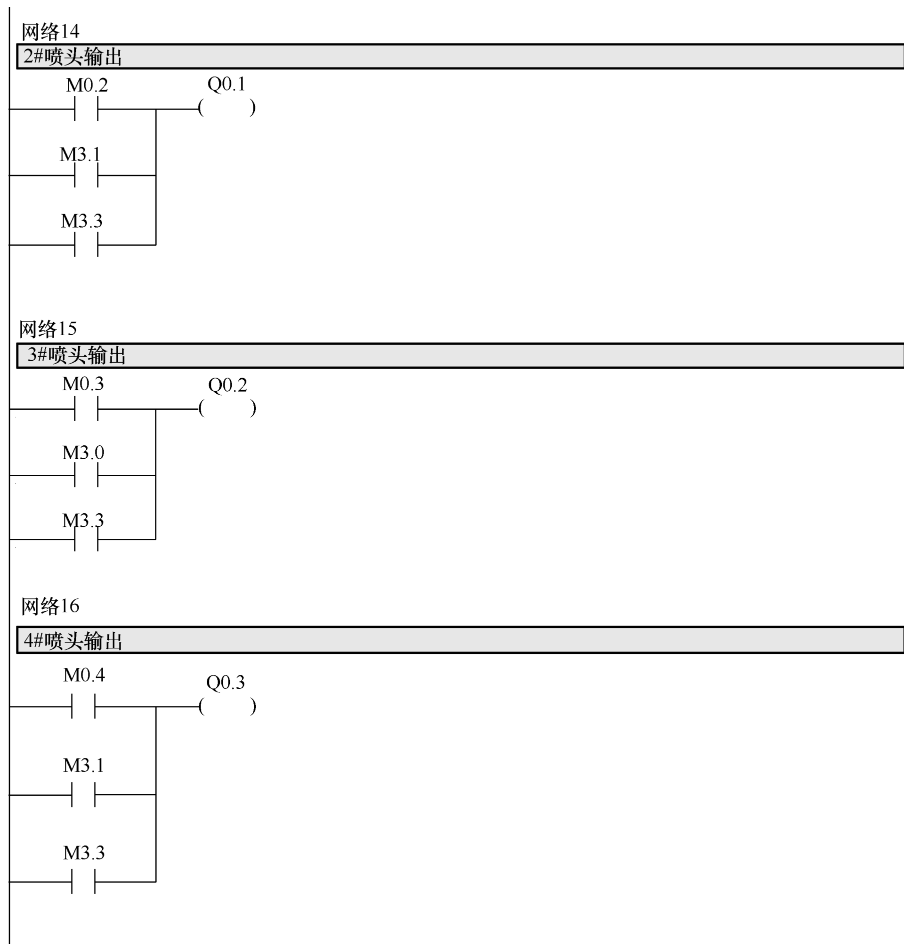


图 4. 12-7　喷泉的 PLC 梯形图程序（续）

实例 4　PLC 与步进电动机的运动控制

1. 控制要求

设计一个简单的步进电动机正反转 PLC 控制系统。按下正转起动按钮 SB1，步进电动机正转；按下反转起动按钮 SB2，步进电动机以相同的转速反转，并转过相同的角度；按下停止按钮 SB3，步进电动机停转。

2. 操作步骤

1）根据控制要求，进行 I/O 地址通道分配。

分析控制要求可知，正转起动按钮 SB1、反转起动按钮 SB2 及停止按钮 SB3 属于控制信号，作为 PLC 的输入量分配接线端子；驱动器属于被控对象，作为 PLC 的输出量分配接线端子。对输入量/输出量（I/O）进行地址分配，见表 4. 12-4。

2）根据控制要求，设计并绘制 PLC 控制系统电气原理图，如图 4. 12-8 所示。PLC 采用晶体管输出型 S7-200 CPU226（DC/DC/DC），步进电动机驱动器控制信号电压为 DC 5V。

表 4.12-4　I/O 地址通道分配

输　入　量			输　出　量		
名　　称	字母代号	地　　址	名　　称	字母代号	地　　址
正转起动按钮	SB1	I0.0	驱动器步进脉冲	PU +	Q0.0
反转起动按钮	SB2	I0.1	驱动器方向控制	DR +	Q0.1
停止按钮	SB3	I0.2			

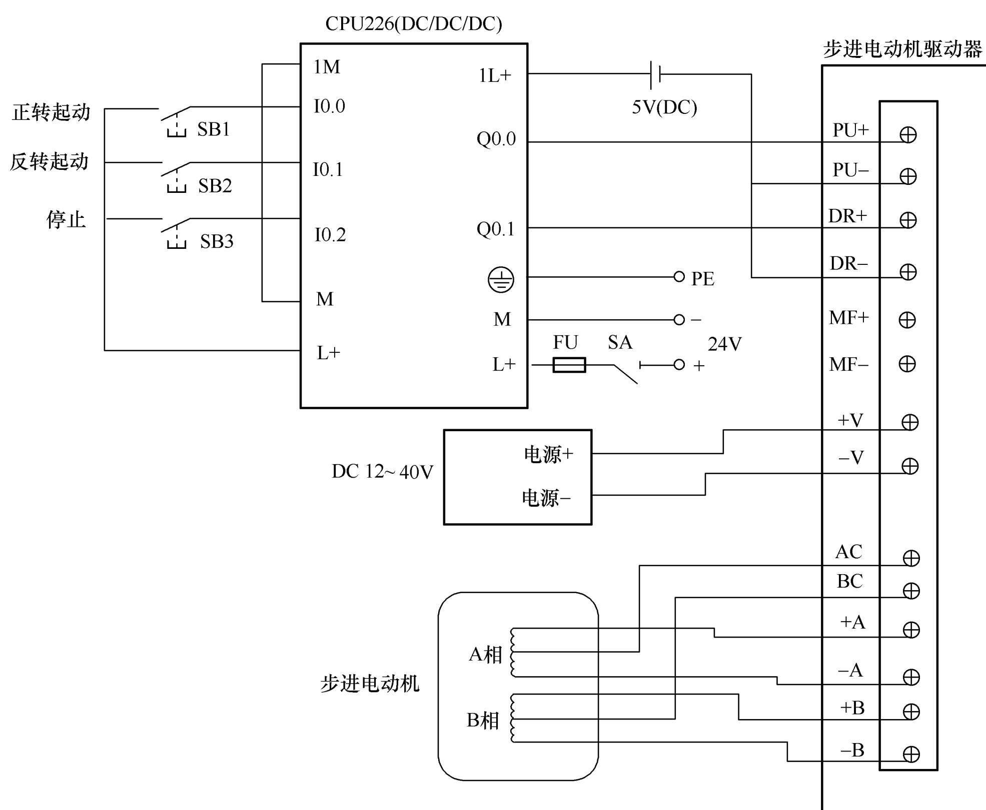
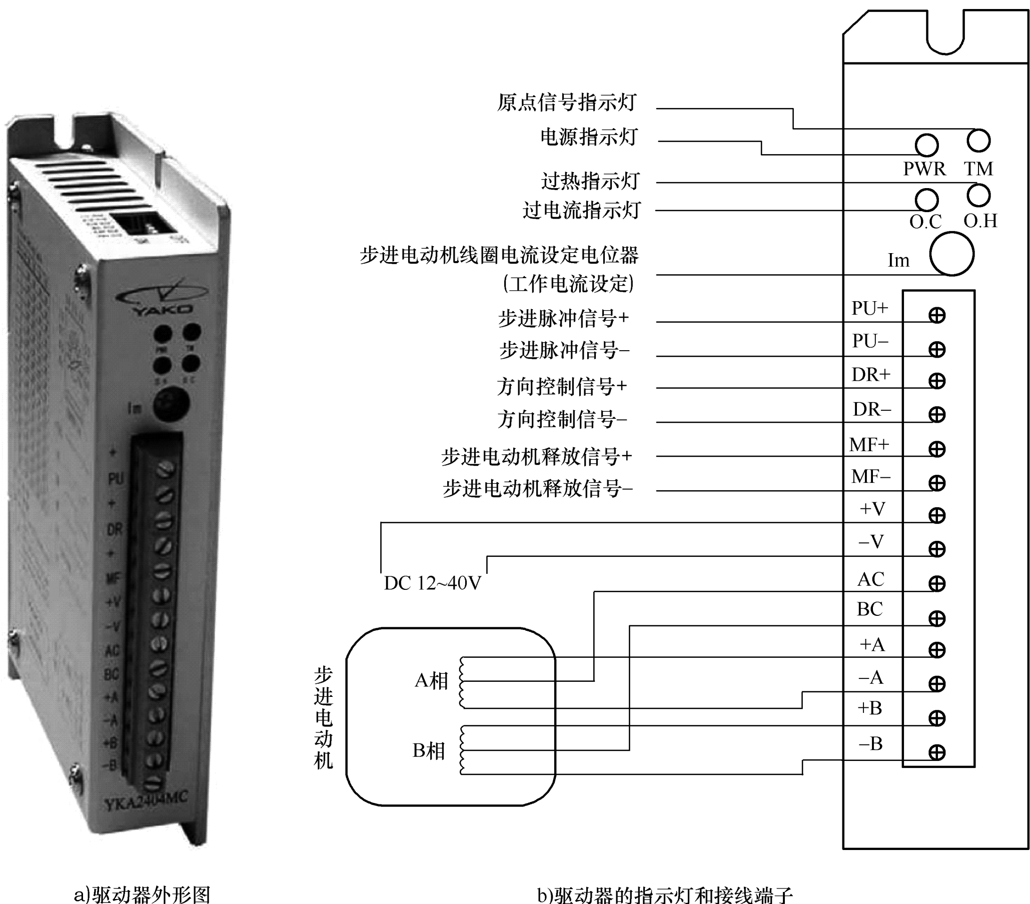


图 4.12-8　PLC 控制步进系统原理图

① YKA2404MC 型驱动器介绍。图 4.12-9a 所示为 YKA2404MC 型驱动器。其工作电压为 DC 12 ~40V，额定电流为 4.0A，采用单电源供电，适配外径尺寸为 42 ~86mm 的各种型号的二相混合式步进电动机。

② YKA2404MC 步进电动机驱动器的接线。YKA2404MC 步进电动机驱动器的指示灯和接线端子如图 4.12-9b 所示，驱动器接线图如图 4.12-8 所示。如把 5V 直流电源脉冲加至 PU + 端与 PU − 端，即把控制器（如 PLC 或单片机等）输出的脉冲信号送至步进电动机驱动器，步进电动机驱动器就按此脉冲的频率去控制步进电动机的转速。DR + 端与 DR − 端用来控制步进电动机的转动方向，此两端子未加上 5V 的直流电压，步进电动机转动方向为正转；两端子上加上 5V 的直流电压，则步进电动机转动方向变为反转。MF 用来控制步进电动机制动。



a)驱动器外形图　　b)驱动器的指示灯和接线端子

图 4.12-9　驱动器介绍

YKA2404MC 步进电动机驱动器的细分设定开关 D2 设置为 OFF，即 PU－端为步进脉冲信号端，DR－端为方向控制信号端。即驱动器的 PU－端和 DR－端与电源的负极连接，驱动器的 PU＋端和 DR＋端应该分别与 PLC 的 Q0.0 和 Q0.1 连接。这样，＋5V 的步进脉冲信号由 Q0.0 端输出通过 PU＋端输入驱动器，＋5V 的方向控制信号由 Q0.1 端输出通过 DR＋端输入驱动器。

3）材料准备。根据接线原理图，列出需要的所有材料清单，见表 4.12-5。

① 选择元件时，主要考虑元件的数量、型号及额定参数，驱动器与步进电动机应相互匹配。

② 检测元器件的质量好坏。

③ PLC 的选型要合理，在满足要求下尽量减少 I/O 的点数，以降低硬件的成本。

4）线路安装与接线。根据控制要求与电路原理图选择安装所需要的材料，并按照工艺要求进行安装与接线。这里重点介绍步进电动机驱动器使用注意事项。

① 不要将电源接反，输入电压不要超过 DC 40V。

② 输入控制信号电平为 5V，当高于 5V 时需要接限流电阻。

③ YKA2404MC 驱动器采用特殊的控制电路，故必须使用 6 出线或者 8 出线步进电动机。

表 4.12-5　材料清单

序　　号	分　　类	名　　称	型号规格	数　　量	备　　注
1	工具	电工工具	自定	1 套	
2	器材	万用表	MF47 型	1 块	
3		可编程序控制器	S7-200/226　DC/DC/DC	1 台	
4		计算机	自定	1 台	
5		S7-200 编程软件	STEP7-Micro/WIN V4.0 SP3	1 套	
6		配电盘	500mm×600mm	1 块	
7		导轨	C45	0.5m	
8		自动断路器	DZ47-63/2P D15 DZ47-63/2P D10	1 只 1 只	
9		步进驱动器	YKA2404MC	1 个	
10		二相混合式步进电动机	自定	1 台	
12		12～40V 直流电源	自定	1 个	
13		直流 24V/5V 电源	自定	1 个	
14		按钮	LA4-3H	1 只	
15		端子排	D-20	1 根（20 节）	
16		主令开关	LS2-2	1 只	
4-9-		熔断器	RT28-32	1 只	
19	耗材	铜塑线	BVR/1.5mm²	10m	主电路
20		铜塑线	BVR/0.5mm²	25m	控制电路
21		紧固件	螺钉（型号自定）	若干	
22		线槽	25mm×35mm	若干	
23		号码管		若干	

④ 驱动器温度超过 70℃时停止工作，故障指示灯 O.H 亮，直到驱动器温度降到 50℃，驱动器自动恢复工作。出现过热保护请加装散热器。

⑤ 过电流（电流过大或电压过小）时故障指示灯 O.C 亮，请检查步进电动机接线及其他短路故障或是否电压过低。若是步进电动机接线及其他短路故障，排除后需要重新上电恢复。

⑥ 驱动器通电时绿色指示灯 PWR 亮。

⑦ 过零点时，TM 指示灯在脉冲输入时亮。

5）PLC 程序设计。编写思路：正转运行时 Q0.0 输出脉冲，Q0.1 为 OFF；反转运行时 Q0.0 输出脉冲，Q0.1 为 ON；停止运行时 Q0.0 停止输出脉冲，Q0.1 为 OFF。由于步进电动机正、反转的转速相同，并转过相同的角度，因此步进电动机正转和反转所需要的脉冲串的脉冲周期和脉冲数一样，即步进电动机正、反转所需要的脉冲串的控制参数一样。由于只需要输出两个脉冲串，可以采用单段 PTO 编程实现控制即可。图 4.12-10 为采用 PLC 脉冲输出指令设计的步进电动机正反转 PLC 控制梯形图。

6）程序录入与调试。

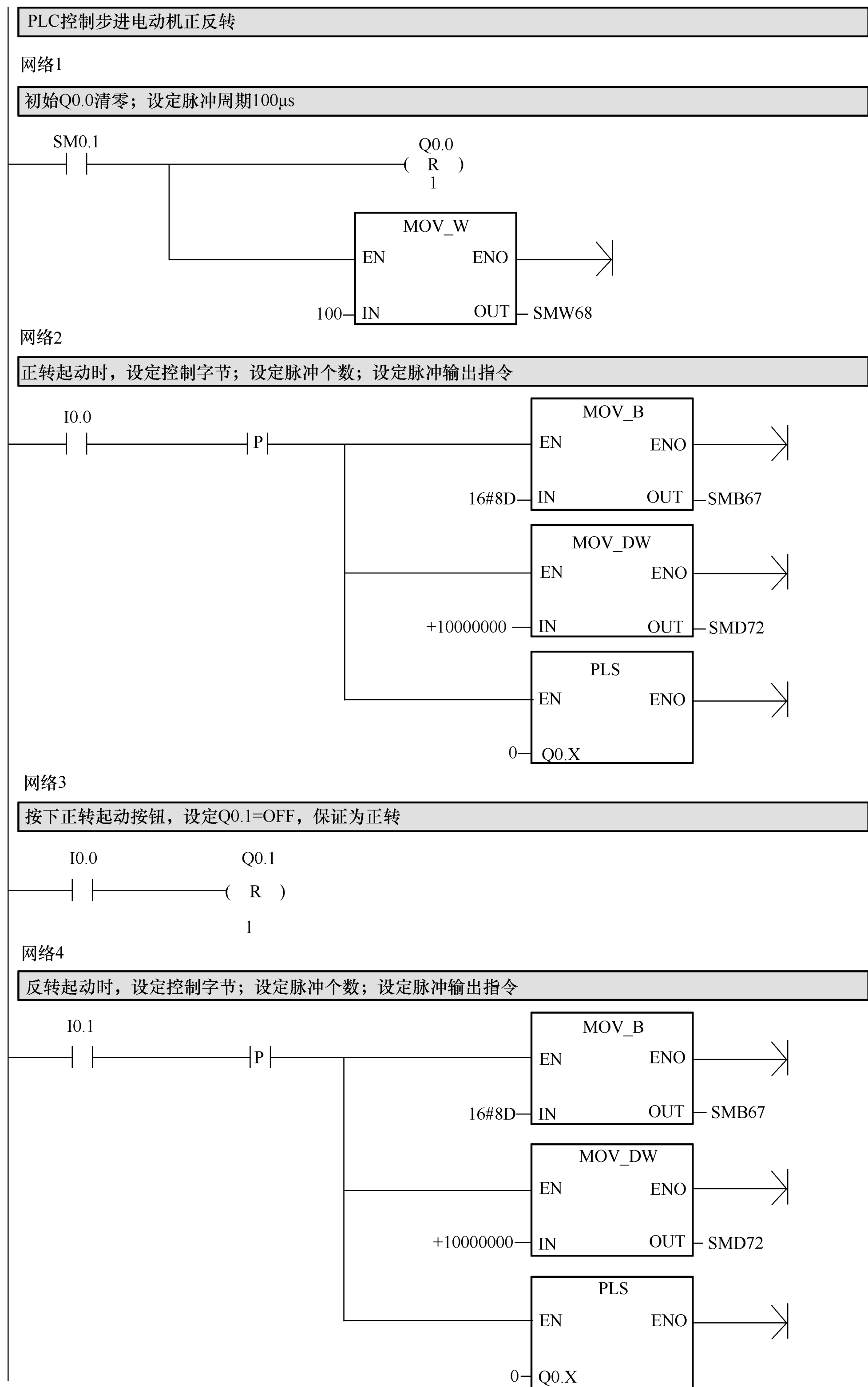


图 4. 12-10　步进电动机正反转 PLC 控制梯形图

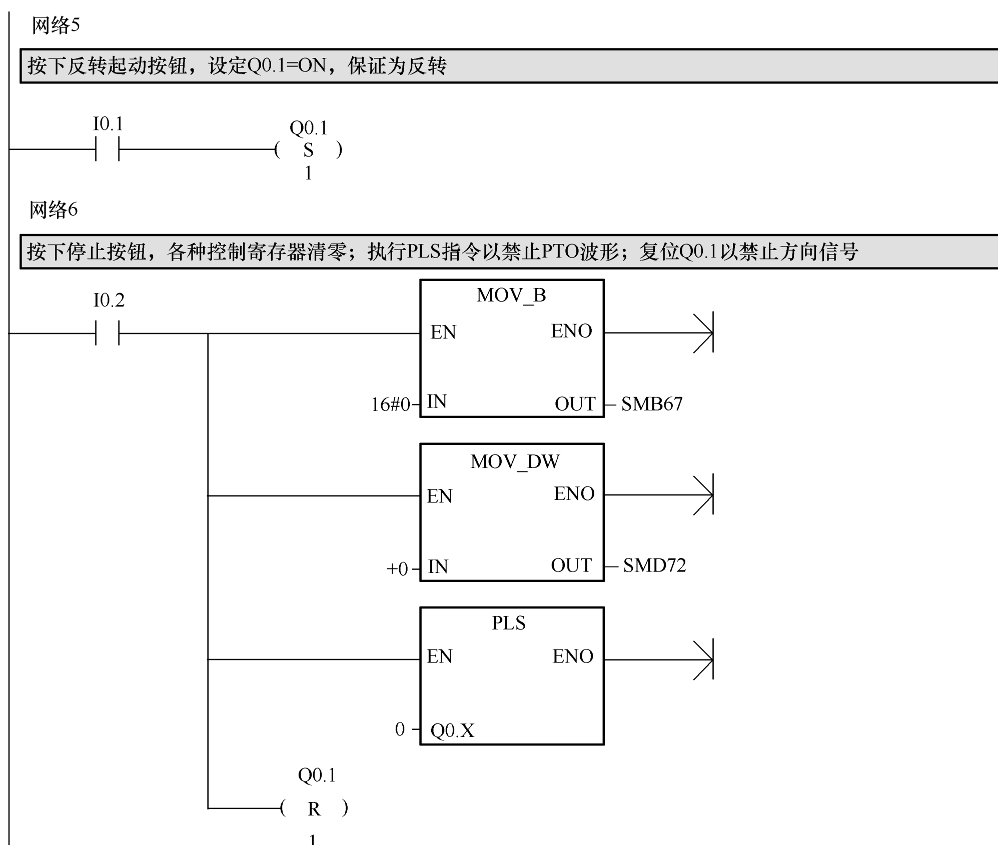


图 4.12-10　步进电动机正反转 PLC 控制梯形图（续）

① 建立计算机与 PLC 通信联系。

② 编译、下载程序。

③ 设置步进驱动器参数。

a. 设定细分数。参照表 4.12-6，使用步进驱动器上的细分设定开关来设定细分数。在系统频率允许的情况下，尽量选用高细分数。

表 4.12-6　YKA2404MC 步进电动机驱动器细分设定表

细 分 数	1	2	4	5	8	10	20	25	40	50	100	200	200	200	200	200
D6	ON	OFF	ON	OFF	ON	OFF	ON	OFF	ON	OFF	ON	OFF	ON	OFF	ON	OFF
D5	ON	ON	OFF	OFF	ON	ON	OFF	OFF	ON	ON	OFF	OFF	ON	ON	OFF	OFF
D4	ON	ON	ON	ON	OFF	OFF	OFF	OFF	ON	ON	ON	ON	OFF	OFF	OFF	OFF
D3	ON	ON	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF
D2	ON，双脉冲：PU 为正向步进脉冲信号，DR 为反向步进脉冲信号 OFF，单脉冲：PU 为步进脉冲信号，DR 为方向控制信号															
D1	无效															

b. 设定步进电动机工作电流。使用步进驱动器上的电动机线圈电流设定电位器来设定

步进电动机的工作电流。顺时针旋转电位器，工作电流减小；逆时针旋转电位器，工作电流增大。

④ 运行与调试程序。

a. 按下正转起动按钮 SB1，Q0.0 输出脉冲，步进电动机以设定的脉冲周期和个数来正转运行。

b. 按下反转起动按钮 SB2，Q0.1 输出脉冲，步进电动机以设定的脉冲周期和个数来反转运行。

c. 按下停止按钮 SB3，步进电动机停止运行。

第五章

PLC 网络通信技术应用

实例 1　两台 S7-200 PLC 的 PPI 通信

1. 控制要求

两台 S7-200 PLC 通过 PORT0 口实现 PPI 通信，功能是由 A 机（2 号主站）的 IB0 控制 B 机（3 号从站）的 QB0，由 B 机的 IB0 控制 A 机的 QB0。

2. 相关知识

（1）PPI 协议是点对点接口的一个主/从协议　主站向从站发送申请，从站进行响应，从站不发信息，不初始化信息，只是等待主站的要求并对要求做出响应。主站设备通过由 PPI 协议管理的共享连接与从站设备通信。PPI 通信协议不限制能够与任何一台从站设备通信的主站设备数量，但在硬件上整个网络中安装的主站设备必须少于 32 台。PPI 协议是 S7-200 最基本的通信方式，通过自身的通信口（PORT0、PORT1）支持 PPI 通信协议，网络中的所有 S7-200 都默认为从站。

S7-200 CPU 之间的 PPI 网络通信只需要两条简单的指令，它们是网络读（NETR）和网络写（NETW）指令，具体介绍见表 5.1-1。在网络读写通信中，只有主站需要调用 NETR/NETW 指令，从站只需编程处理数据缓冲区（取用或预备数据）。实现 PPI 网络读写通信，可有两种方法：①使用 NETR/NETW 指令实现编程；②使用 Micro/WIN 中的指令向导 NETR/NETW（见实例 2）。PPI 网络上的所有站点都应当有各自不同的网络地址，所有 PLC 的通信速率必须相同，否则通信不成功。

表 5.1-1　网络通信指令

指令名称	梯形图	指　令	功　能	数据类型及操作数
网络读指令	NETR EN ENO TBL PORT	NETR TBL，PORT	当使能输入 EN 有效时，通过 PORT 指定的通信口（主站上的 0 或 1），根据 TBL 指定的表中的定义读取远程装置的数据	数据类型：字节型 TBL：VB、MB、*VD、*AC、*LD PORT：0 或 1 的常数
网络写指令	NETW EN ENO TBL PORT	NETW TBL，PORT	当使能输入 EN 有效时，通过 PORT 指定的通信口（主站上的 0 或 1），根据 TBL 指定的表中的定义将数据写入远程设备中	数据类型：字节型 TBL：VB、MB、*VD、*AC、*LD PORT：0 或 1 的常数

（2）网络读写指令　网络读写指令可以向远程站发送或接收 16 个字节的信息，在 CPU

内可同时激活 8 条指令。网络读写指令是通过 TBL 参数来指定的。功能图中 TBL 处的字节是数据表的起始地址，可以由用户自己定，但起始字节定好后，后面的字节就要连续使用，形成列表，每字节都有具体的任务。表 5. 1-2 所示为数据表（TBL）格式。

表 5. 1-2　数据表格式

字　　节	bit7				bit0
0	D	A	E	0	错误代码
1	远程站地址				
2	远程站的数据指针 （I、Q、M、V）				
3					
4					
5					
6	信息字节总数				
7	信息字节 0				
8	信息字节 1				
⋮	⋮				
22	信息字节 15				

注：
D 表示操作完成状态，0 = 未完成；1 = 已完成。
A 表示操作有效否，0 = 无效；1 = 有效。
E 错误信息，0 = 无错；1 = 有错。

错误代码见表 5. 1-3。

表 5. 1-3　错误代码

错误代码	表示意义
0	没有错误
1	远程站响应超时
2	接收错误：奇偶校验错，响应时帧或校验错
3	离线错误：相同的站地址或无效的硬件引发冲突
4	队列溢出错误：同时激活超过 8 条网络读写指令
5	通信协议错误：没有使用 PPI 协议而调用网络读写指令
6	非法参数
7	远程站正在忙
8	第七层错误：违反应用协议
9	错误信息：数据地址或长度错误
10	保留

SMB30 和 SMB130 分别是 S7-200 PLC PORT0 及 PORT1 通信口的控制字节，各位的意义见表 5. 1-4。

表 5.1-4　SMB30 和 SMB130 各位控制字节的意义

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
p	p	d	b	b	b	m	m

pp：校验选择	d：每个字符的数据位	mm：协议选择
00 = 不校验	0 = 8 位	00 = PPI/从站模式
01 = 偶校验	1 = 7 位	01 = 自由口模式
10 = 不校验		10 = PPI/主站模式
11 = 奇校验		11 = 保留（未用）
bbb：自由口波特率	（单位：Baud）	
000 = 38400	011 = 4800	110 = 115.2k
001 = 19200	100 = 2400	111 = 57.6k
010 = 9600	101 = 1200	注：查看 CPU 版本

（3）网络通信硬件　S7-200 支持的 PPI、PROFIBUS-DP 和自由口通信模式都是建立在 RS485 的硬件基础上的，为了保证足够的传输距离和通信速率，建议使用西门子制造的网络电缆和网络连接器，如图 5.1-1 所示。

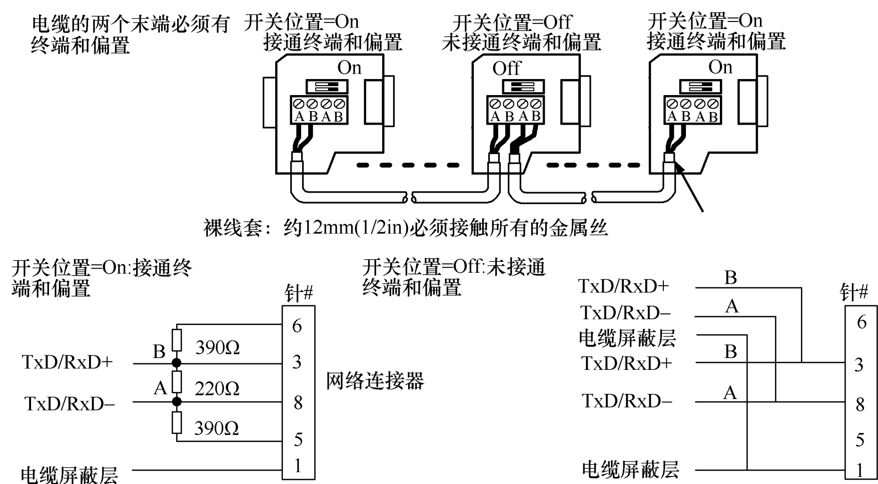


图 5.1-1　网络电缆和网络连接器

3. 操作步骤

（1）网络配置和 I/O 的分配

PPI 协议是 S7-200 PLC 比较简单的一种通信方式，图 5.1-2 为两台 S7-200 PLC 实现 PPI 通信的网络配置图，I/O 分配见表 5.1-5。

表 5.1-5　I/O 分配

2 号主站（S7-200）	3 号从站（S7-200）
2 号站 IB0 控制 3 号站 QB0	3 号站 IB0 控制 2 号站 QB0

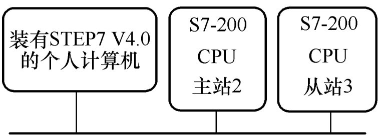


图 5.1-2　网络配置图

(2) 通信端口设置

在输入程序之前先进行通信端口设置，步骤如下：

1) 打开 STEP7 V4.0 编程软件，如图 5.1-3 所示，选择“系统块”，打开“通讯端口”。



图 5.1-3　编程软件界面

2) 设置 PLC 端口 0 的地址为“3”，选择波特率为“9.6 千波特”，如图 5.1-4 所示；然后把系统参数下载到 CPU 中，如图 5.1-5 所示。用同样的方法设置另一个 CPU，端口 0 的地址为“2”，波特率为“9.6 千波特”，同样把参数下载到 CPU 中。

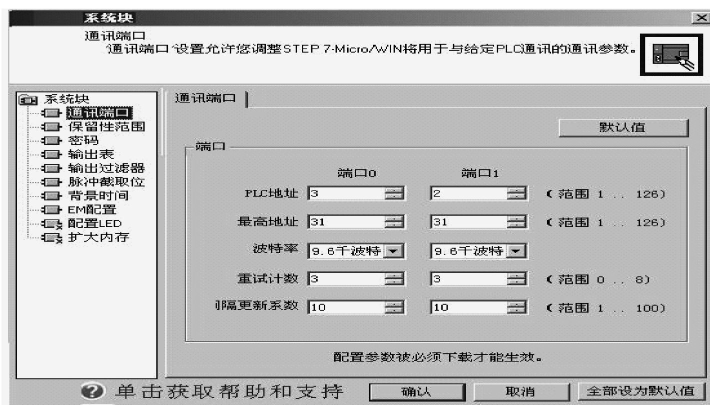


图 5.1-4　设置通信端口



图 5.1-5　下载参数

3）利用网络插头及网络线把 A 机和 B 机端口 0 相连接，利用软件搜索，如图 5.1-6 所示。



图 5.1-6　PPI 网络中的 S7-200 站

（3）梯形图程序

图 5.1-7 所示为 A 机主站梯形图程序，在主站中输入此梯形图程序。

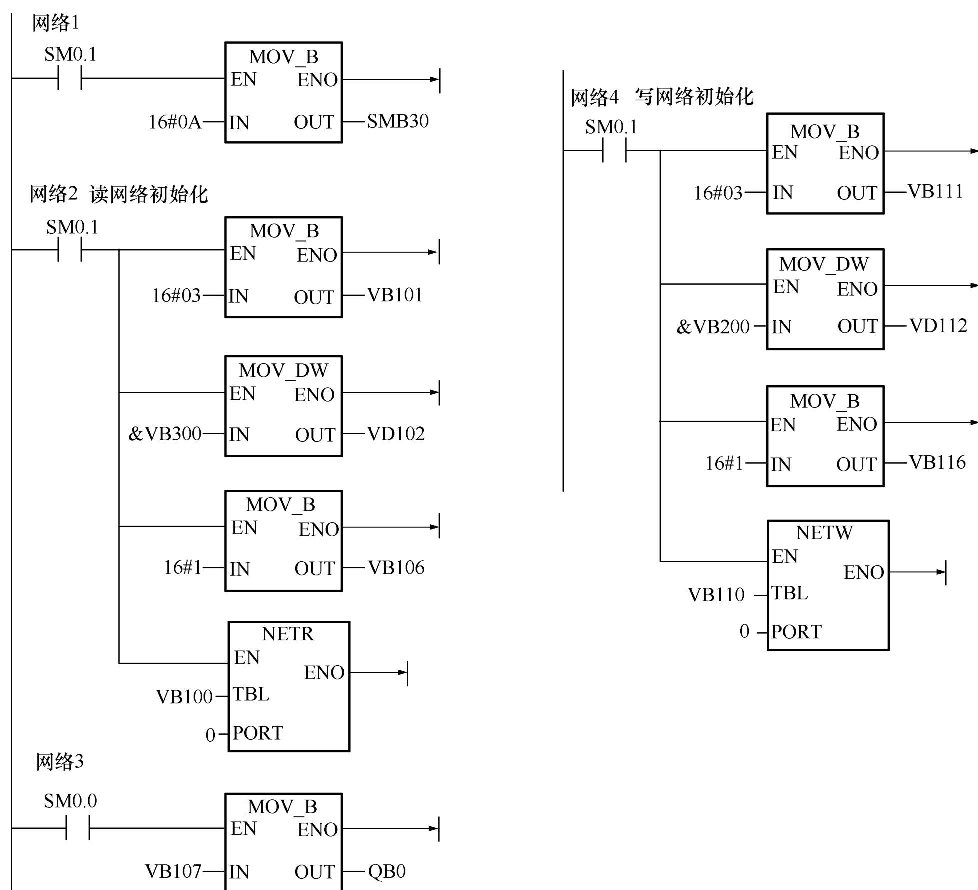


图 5.1-7　A 机主站梯形图程序

图 5. 1-8 所示为 B 机从站梯形图程序，在从站中输入此梯形图程序。

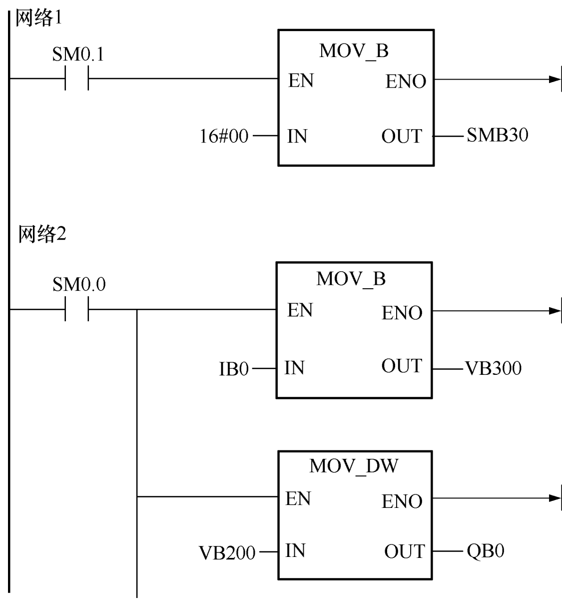


图 5. 1-8　B 机从站梯形图程序

（4）接线与调试

用一根 RS485 通信线连接主站和从站的 PORT0 端口，如图 5. 1-9 所示；然后分别接通主站或从站的输入端 IB0，观察主站或从站的输出端 QB0 的状态，直到调试成功。

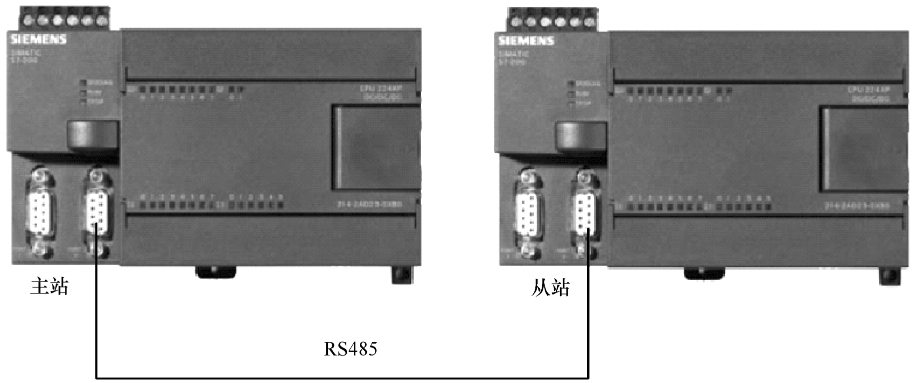


图 5. 1-9　S7-200 PLC PPI 通信连接图

实例 2　S7-200 PLC 指令向导编程的 PPI 通信

1. 控制要求

本实例要求将主站的 I1. 0 ~ I1. 7 的状态映射到从站的 Q0. 0 ~ Q0. 7，将从站的 I1. 0 ~ I1. 7 的状态映射到主站的 Q0. 0 ~ Q0. 7。

2. 操作步骤

（1）NETR/NETW 指令向导设置

1）启动 STEP7- Micro/WIN 编程软件，在指令树中，单击“向导”找到“NETR/NETW”指令向导并双击打开，如图 5.2-1 所示，选择配置网络读/写操作为“2”。

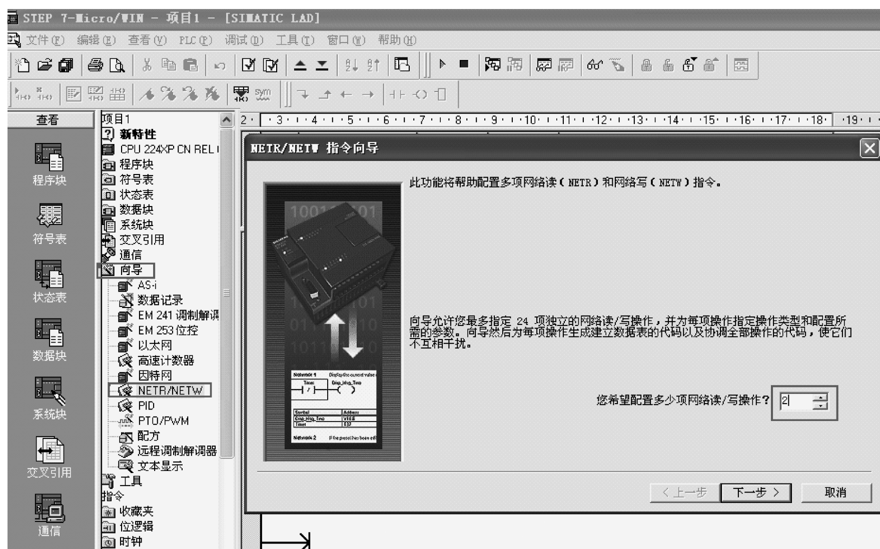


图 5.2-1　NETR/NETW 指令向导设置 1

2）单击图 5.2-1 中的“下一步”按钮，选择 PLC 的通信端口 0，子程序名称默认为“NET_EXE”如图 5.2-2 所示。

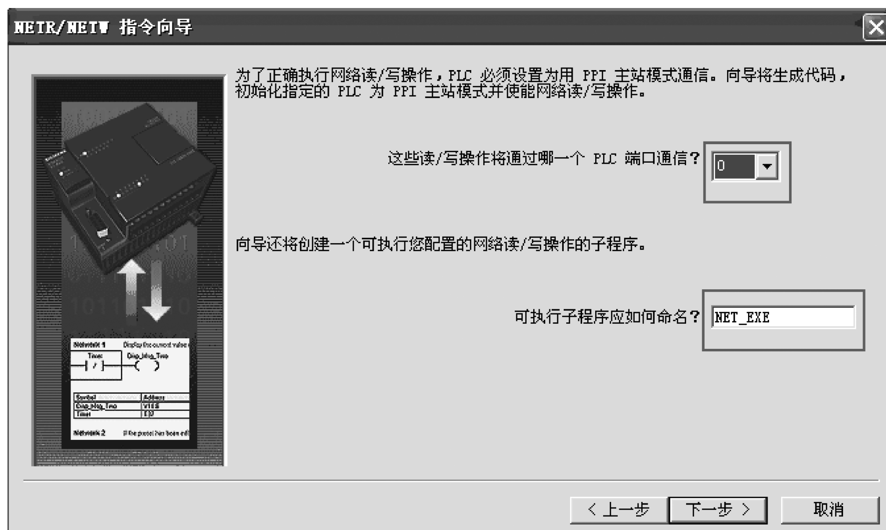


图 5.2-2　NETR/NETW 指令向导设置 2

3）单击图 5.2-2 中的“下一步”按钮，出现网络读/写操作第 1 项界面，如图 5.2-3 所示。在第 1 项操作设为“NETR”网络读操作；读取字节数为“1”；远程 PLC 地址设为

“6”；数据传输本地存储设为“VB307 至 VB307”、远程读取地址设为“VB200 至 VB200”。

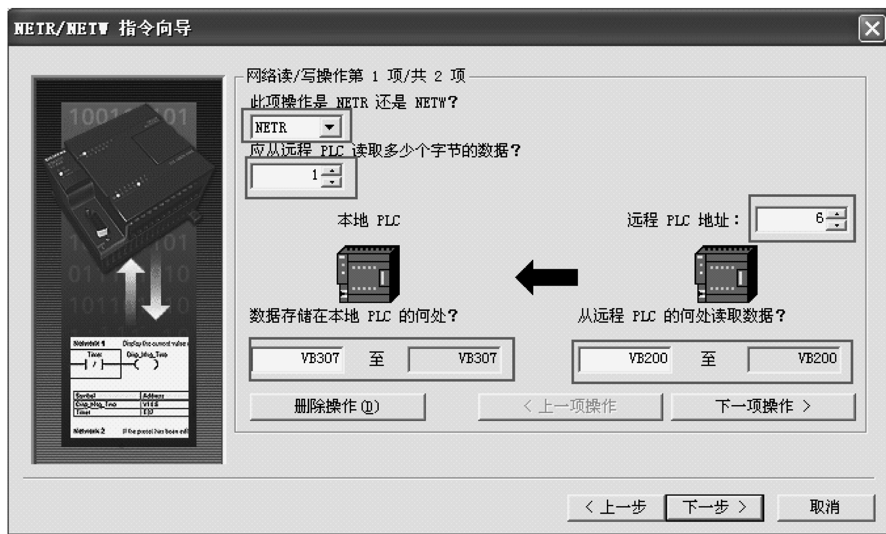


图 5.2-3　NETR/NETW 指令向导设置 3

4）单击图 5.2-3 中的“下一项”按钮，进入网络读/写操作第 2 项界面，如图 5.2-4 所示。设置网络为“NETW”写操作；读取字节数为“1”；远程 PLC 地址设为“6”；数据本地传输地址为“VB207 至 VB207”、远程读取地址为“VB200 至 VB200”。

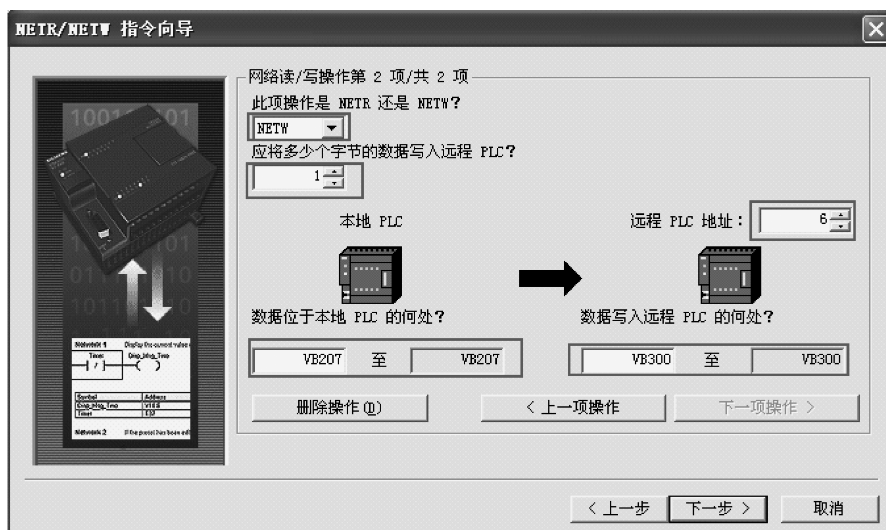


图 5.2-4　NETR/NETW 指令向导设置 4

5）单击图 5.2-4 中的“下一步”按钮，出现分配存储区界面，如图 5.2-5 所示。采用建议地址为“VB19 至 VB37”即可，单击“下一步”按钮进入完成配置向导的组态，如图 5.2-6 所示。

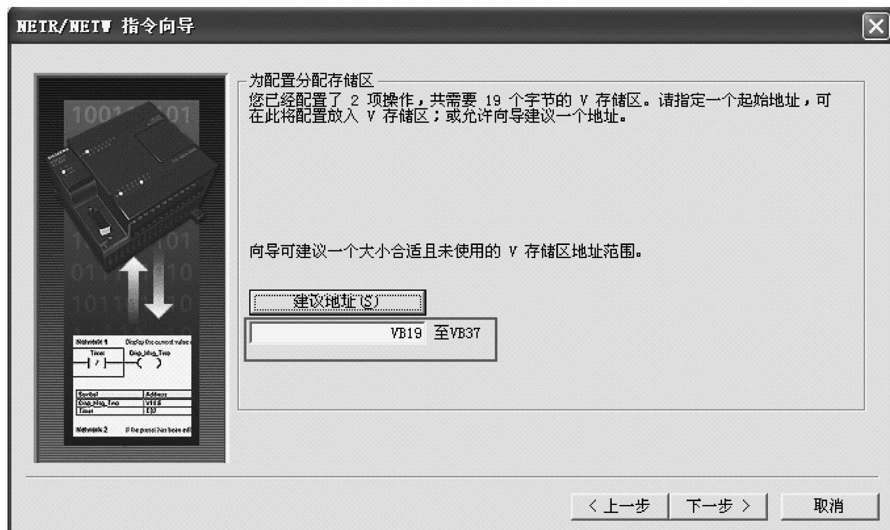


图 5.2-5　NETR/NETW 指令向导设置 5



图 5.2-6　NETR/NETW 指令向导设置 6

（2）编写梯形图程序

1）主站梯形图程序如图 5.2-7 所示。主站程序编写完毕后，把程序下载到主站 PLC 中。

2）从站梯形图程序。从站程序编写完毕后，把程序下载到从站 PLC 中。

首先打开编程软件，并双击“系统块”，将 PLC 端口 0 设置为“6”，如图 5.2-8 所示；然后单击“确认”按钮，进入软件编辑区编辑程序，从站梯形图程序如图 5.2-9 所示。

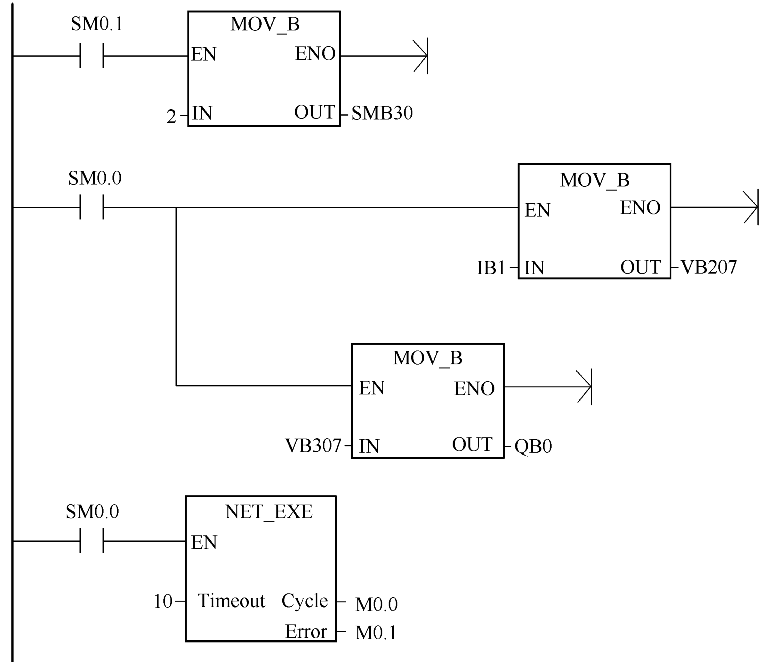


图 5. 2-7 主站主程序

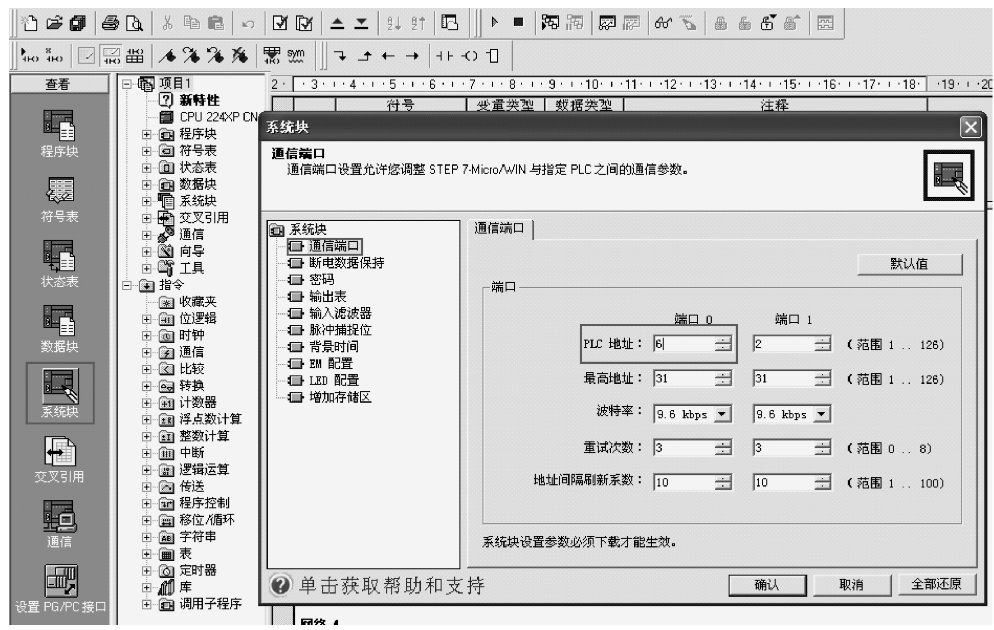


图 5. 2-8 系统块参数设置

（3）接线与调试

用一根 RS485 通信线连接主站和从站的 PORT0 端口，如图 5. 2-9 所示；然后分别接通主站或从站的输入端 IB1，观察主站或从站的输出端 QB0 的状态，直到调试成功。

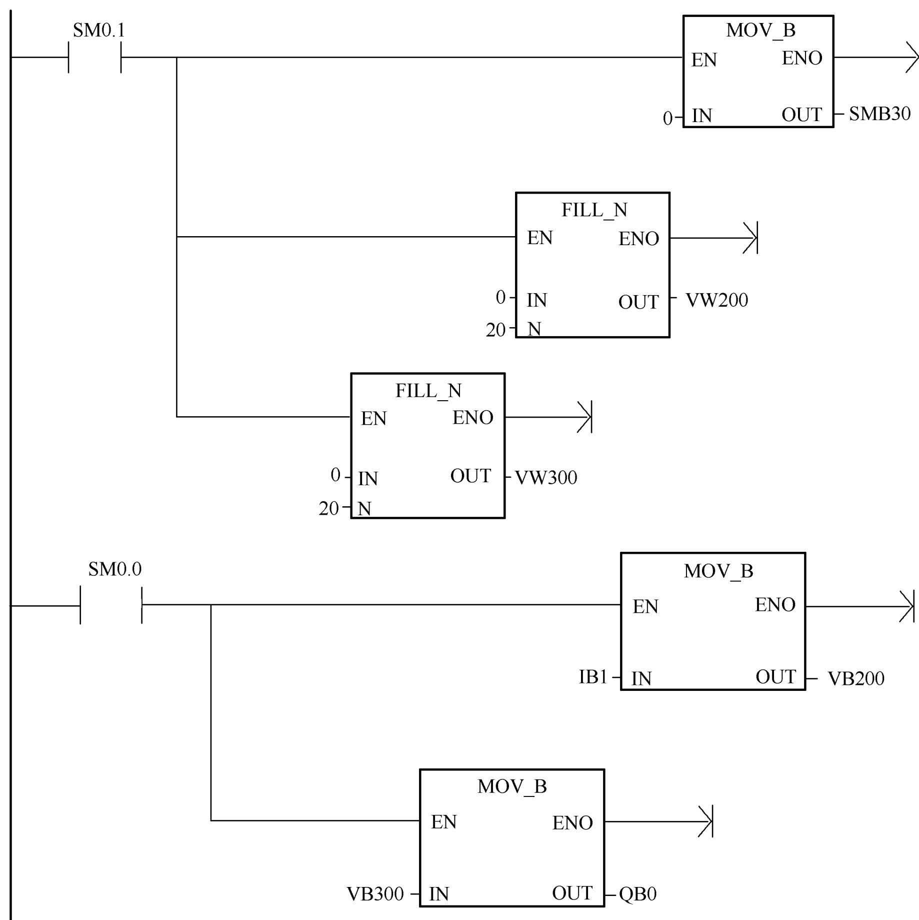


图 5.2-9　从站梯形图程序

实例 3　两台 S7-200 PLC 的 Modbus 通信

1. 控制要求

两台 S7-200 PLC 通过 Modbus 进行通信，一台组态为主站，一台为从站。当主站 I0.2 为 ON 时，通过 Modbus 方式读取从站 S7-200 PLC 的 I0.0 ~ I0.7 的数值。主从站通过 S7-200 的串口编程电缆进行连接。

2. 相关知识

Modbus 通信协议是 Modicon 公司提出的一种报文传输协议，广泛用于工业控制领域，并已经成为一种通用的行业标准。Modbus 通信协议有 ASCII 和 RTU 两种报文传输模式。在使用 Modbus 的网络中所有的站必须采用相同的传输模式和串口参数。目前，西门子公司专门针对西门子 PLC Modbus RTU 通信开发了指令库，极大简化了 Modbus RTU 通信的开发，以便快速实现相关应用。

Modbus 通信指令库包括主站协议指令和从站协议指令。使用 Modbus 协议的要求如下：

1）初始化 Modbus 从属协议为 Modbus 从属协议通信指定端口 0。

当端口 0 用于 Modbus 从属协议通信时，就无法用于任何其他用途，包括与 STEP7-Micro/WIN 通信。MBUS_INIT 指令控制将端口 0 指定给 Modbus 从属协议 PPI。

2）Modbus 从属协议指令影响与端口 0 中自由端口通信相关的所有 SM 位置。

3）Modbus 从属协议指令使用 3 个子例行程序和 2 个中断例行程序。

4）Modbus 从属协议指令要求 2 个 Modbus 从属指令支持例行程序有 1857 个字节的程序空间。

5）Modbus 从属协议指令的变量要求 779 个字节的 V 内存程序块。该程序块的起始地址由用户指定，专门保留用于 Modbus 变量。

（1）Modbus 主站协议指令

西门子 Modbus 主站协议指令库包括两条主站协议指令：MBUS_CTRL 和 MBUS_MSG 指令。

1）MBUS_CTRL 指令。MBUS_CTRL 指令用于主站通信初始化，如图 5.3-1 所示。此指令主要对 S7-200 端口 PORT0 使用。在每次扫描且 EN 使能有效时调用此指令，当正确执行完毕，立即设定“完成”位后，才能继续执行下一条指令。一般使用 SM0.0 调用 MBUS_CTRL 指令完成主站的初始化，并启动其功能控制。MBUS_CTRL 指令的具体参数含义如下。

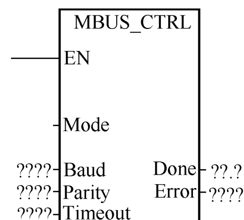


图 5.3-1　MBUS_CTRL 指令

EN：使能位。必须保证每一扫描周期都被使能（使用 SM0.0）。

Mode：模式参数。输入数值来选择通信协议：为 1 时，使能 Modbus 协议功能；为 0 时，恢复为系统 PPI 协议并禁止用 Modbus 协议。

Baud：波特率参数。支持的通信波特率为 1200、2400、4800、9600、19 200、38 400、57 600、115 200。

Parity：奇偶校验参数。奇偶校验参数设置必须与从站的奇偶校验相匹配，校验方式选择：0 = 无校验，1 = 奇校验，2 = 偶校验。

Timeout：超时参数。主站等待从站响应的时间，以 ms 为单位，典型的设置值为 1000ms（1s），允许设置的范围为 1 ~ 32 767。注意：这个值必须设置足够大以保证从站有时间响应。

Done：完成标志位。MBUS_CTRL 指令初始化完成后，此位会自动置 1，否则为 0。可以用该位启动 MBUS_MSG 读写操作。

Error：错误代码。初始化错误代码（只有在 Done 位为 1 时有效）：0 = 无错误，1 = 校验选择非法，2 = 波特率选择非法，3 = 模式选择非法。

MBUS_CTRL 指令参数支持的操作数和数据类型见表 5.3-1。

表 5.3-1　MBUS _CTRL 指令参数支持的操作数和数据类型表

输入/输出	操　作　数	数 据 类 型
Mode	I、Q、M、S、SM、T、C、V、L	布尔
Baud	VD、ID、QD、MD、SD、SMD、LD、AC、Constant、*VD、*AC、*LD	双字
Parity	VB、IB、QB、MB、SB、SMB、LB、AC、Constant、*VD、*AC、*LD	字节
Timeout	VW、IW、QW、MW、SW、SMW、LW、AC、Constant、*VD、*AC、*LD	字
Done	L、Q、M、S、SM、T、C、V、L	布尔
Error	VB、IB、QB、MB、SB、SMB、LB、AC*、VD*、AC*、LD	字节

2）MBUS_MSG 指令。MBUS_MSG 指令用于启动 Modbus 从站的请求并处理应答，如图 5.3-2 所示。当 EN 输入参数和 First（首次）输入参数都为 1 时，MBUS_MSG 指令启动对 Modbus 从站的请求。在发送请求、等待应答并处理应答时通常需要多次扫描，因此 EN 输入必须打开以启用请求的发送，并应该保持打开直到“完成”位被置位。MBUS_MSG 指令的具体参数含义如下。

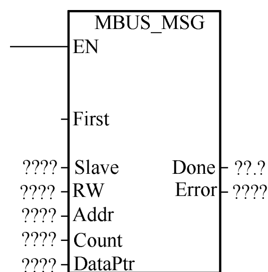


图 5.3-2　MBUS_MSG 指令

EN：使能位。同一时刻只能有一个读写功能（MBUS_MSG）使能。注意：建议每一个读写功能（MBUS_MSG）都用上一个 MBUS_MSG 指令的 Done 完成位来激活，以保证所有读写指令循环进行。

First：首次参数。读写请求位，应在每一个新的读写请求时才打开一次扫描。参数输入必须使用脉冲触发。

Slave：从站地址参数。从站地址参数是 Modbus 从站的地址，可选择的范围是 1～247。

RW：读写参数。读写参数指定是否要读取或写入该消息。参数允许使用两个值：0 = 读，1 = 写。

Addr：地址参数。地址参数是起始的 Modbus 的地址，允许读写的数据地址及类型：

00001～09999 是开关量输出（线圈）；

10001～19999 是开关量输入（触点）；

30001～39999 是模拟量输入（输入寄存器）；

40001～49999 是保持寄存器。

注意：

① 开关量输出和保持寄存器支持读和写功能。

② 开关量输入和模拟量输入只支持读功能。

③ 000001～000128 是映射至 Q0.0～Q15.7 的开关量输出。

④ 010001～010128 是映射至 I0.0～I15.7 的开关量输入。

⑤ 030001～030032 是映射至 AIW0～AIW62 的模拟输入寄存器。

⑥ 040001～04xxxx 是映射至 V 内存的保持寄存器。

Count：计数参数。在该请求中读取或写入数据的个数，计数的数值是位数或字数。注意：Modbus 主站可读/写的最大数据量为 120 个字（是指每一个 MBUS_MSG 指令）。

DataPtr：数据指针参数。只针对 S7-200 的 V 存储器中与读取或写入请求相关的数据的间接地址指针。如果是读指令，读回的数据放到这个数据区中；如果是写指令，要写出的数据放到这个数据区中。

Done：完成参数位。读写功能完成位。

Error：错误代码。只有在 Done 位为 1 时，错误代码才有效。错误代码含义如下：

0 = 无错误

1 = 响应校验错误

2 = 未用

3 = 接收超时（从站无响应）

4 = 请求参数错误（slave address，Modbus address，count，RW）

5 = Modbus/自由口未使能
6 = Modbus 正在忙于其他请求
7 = 响应错误（响应不是请求的操作）
8 = 响应 CRC 校验和错误
101 = 从站不支持请求的功能
102 = 从站不支持数据地址
103 = 从站不支持此种数据类型
104 = 从站设备故障
105 = 从站接收了信息，但是响应被延迟
106 = 从站忙，拒绝了该信息
107 = 从站拒绝了信息
108 = 从站存储器奇偶错误

MBUS_MSG 指令支持的操作数和数据类型见表 5. 3-2。

表 5. 3-2　MBUS_MSG 指令支持的操作数和数据类型

输入/输出	操　作　数	数据类型
First	I、Q、M、S、SM、T、C、V、L（触发脉冲信号）	布尔
Addr	VD、ID、QD、MD、SD、SMD、LD、AC、Constant、*VD、*AC、*LD	双字
Slave	VB、IB、QB、MB、SB、SMB、LB、AC、Constant、*VD、*AC、*LD	字节
RW	VB、IB、QB、MB、SB、SMB、LB、AC、Constant、*VD、*AC、*LD	字节
Count	VW、IW、QW、MW、SW、SMW、LW、AC、Constant、*VD、*AC、*LD	整数
Dataptr	&VB	双字
Done	L、Q、M、S、SM、T、C、V、L	布尔
Error	VB、IB、QB、MB、SB、SMB、LB、AC*、VD*、AC*、LD	字节

（2）Modbus 从站协议指令

西门子 Modbus 从站协议指令库包括两条指令：MBUS_INIT 指令和 MBUS_SLAVE 指令。

1）MBUS_INIT 指令。MBUS_INIT 指令用于启用和初始化或禁止 Modbus 从站通信，如图 5. 3-3 所示。在使用 MBUS_SLAVE 指令之前，必须正确执行 MBUS_INIT 指令，完成位置位后，才能继续执行下一条指令。在每次扫描且 EN 输入打开时执行该指令。其参数含义如下。

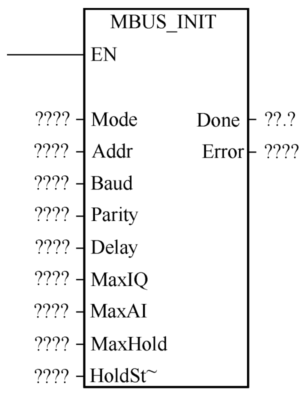


图 5. 3-3　MBUS_INIT 指令

EN：输入使能位。应当在每次通信状态改变时执行 MBUS_INIT 指令。因此，EN 输入应当通过一个边缘脉冲打开，或者仅在首次扫描时执行。

Mode：模式参数。输入参数数值选择通信协议：输入数值 1，将端口 0 指定给 Modbus 协议并启用协议；输入数值 0，将端口 0 指定给 PPI，并禁用 Modbus 协议。

Baud：波特率参数。波特率可选择 1200、2400、4800、9600、19 200、38 400、57 600 或 115 200。

Addr：地址参数。输入 Modbus 从站地址参数，设置范围为 1～247 之间（包括 1 和 247）的数值。

Parity：奇偶校验参数。被设为与 Modbus 主设备奇偶校验相匹配，可设置的数值：0 = 无奇偶校验，1 = 奇数奇偶校验，2 = 偶数奇偶校验。

Delay：延迟参数。通过将指定的毫秒数增加至标准 Modbus 信息超时的方法延长标准 Modbus 信息结束超时条件。该参数的典型数值为 0。如果使用带有纠错功能的调制解调器，将延迟设为 50～100ms 的数值；如果使用扩展频谱无线电，将延迟设为 10～100ms 的数值。“延迟”数值可以是 0～32 767ms。

MaxIQ：最大 I/O 位参数。将 Modbus 地址 00xxxx 和 01xxxx 使用的 I 和 Q 点数设为 0～128 之间的数值。数值 0 禁用所有向输入和输出的读取。建议 MaxIQ 数值是 128，该数值可在 S7-200 中存取所有的 I 和 Q 点。

MaxAI：最大 AI 参数。将 Modbus 地址 03xxx 使用的字输入（AI）寄存器数目设为 0～32 之间的数值。数值 0 禁用模拟输入的读数。建议使用 MaxAI 数值如下，这些数值可允许存取所有的 S7-200 模拟输入：0 用于 CPU 221，16 用于 CPU 222，32 用于 CPU 224、226 和 226XM。

MaxHold：参数设定供 Modbus 地址 04xxx 使用的 V 内存中的字保持寄存器数目。例如，为了允许主设备存取 2000 个字节的 V 内存，将“MaxHold”设为 1000 个字的数值（保持寄存器）。

HoldStart：参数是 V 存储区的保持寄存器的起始地址，该数值一般被设为 VB0，因此参数 HoldStart 设为 &VB0（VB0 的地址）。也可以将其他的 V 区地址指定为保持寄存器的起始地址，以便使 VB0 可以在项目中用作其他目的。Modbus 主站可以访问起始地址为 HoldStart，字数为 MaxHold 的 V 存储区。

Done：初始化完成标志位，成功初始化后置 1。

Error：初始化错误代码。

MBUS_INIT 指令参数支持的操作数和数据类型见表 5.3-3。

表 5.3-3　MBUS_INIT 指令参数支持的操作数和数据类型

输入/输出	操　作　数	数 据 类 型
模式、地址、奇偶校验	VB、IB、QB、MB、SB、SMB、LB、AC、Constant、＊VD、＊AC、＊LD	布尔
波特率、Holdstart	VD、ID、QD、MD、SD、SMD、LD、AC、Constant、＊VD、＊AC、＊LD	双字
延时、MaxIQ、MaxAI、MaxHold	VW、IW、QW、MW、SW、SMW、LW、AC、Constant、＊VD、＊AC、＊LD	字节
Done	L、Q、M、S、SM、T、C、V、L	布尔
Error	VB、IB、QB、MB、SB、SMB、LB、AC＊、VD＊、AC＊、LD	字节

2）MBUS_SLAVE 指令。MBUS_SLAVE 指令用于为 Modbus 主设备发出的请求服务，并且必须在每次扫描时执行，以便允许该指令检查和回答 Modbus。在每次扫描且 EN 输入开启时执行该指令。MBUS_SLAVE 指令无输入参数，如图 5.3-4 所示。

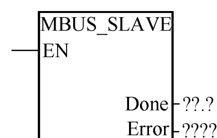


图 5.3-4　MBUS_SLAVE 指令

当 MBUS_SLAVE 指令对 Modbus 请求做出应答时，“完成”输出打开。如果没有需要服务的请求，“完成”输出关闭。“错误”输出包含执行该指令的结果，该输出只有在“完成”打开时才有效。如果“完成”关闭，错误参数不会改变。

MBUS_SLAVE 指令参数支持的操作数和数据类型见表 5.3-4。

表 5.3-4 MBUS_SLAVE 指令参数支持的操作数和数据类型

输入/输出	操作数	数据类型
Done	L、Q、M、S、SM、T、C、V、L	布尔
Error	VB、IB、QB、MB、SB、SMB、LB、AC*、VD*、AC*、LD	字节

3. 操作步骤

（1）Modbus 主站组态

1）分配库存储区。打开编程软件选择“文件”/“库存储区”，出现如图 5.3-5 所示的界面，单击“建议地址”按钮系统自动分配存储区的地址。

2）主站编程。Modbus 主站指令编程如图 5.3-6 所示。

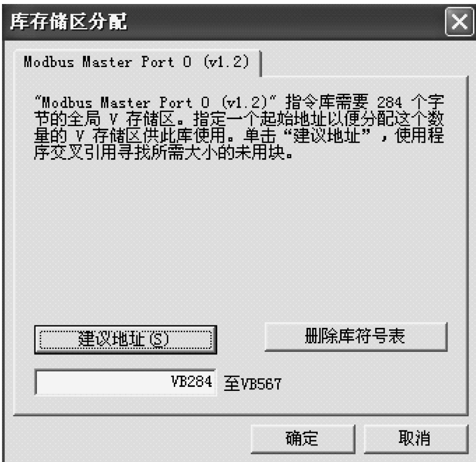


图 5.3-5 分配存储区

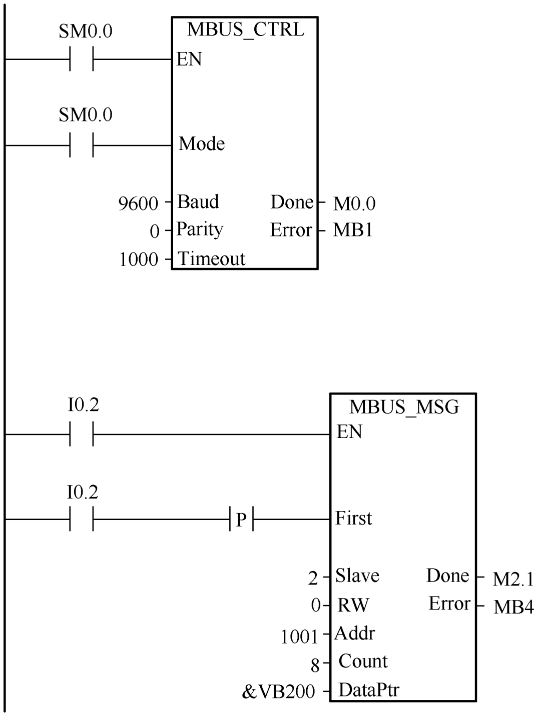


图 5.3-6 主站主程序

（2）Modbus 从站组态

1）分配库存储区。分配库存储区与主站类似。

2）从站编程。Modbus 从站指令编程如图 5.3-7 所示。

（3）通信调试

用串口编程电缆连接主从站 PLC 的端口 0，设主从站为运行状态，并接通 I0.2。从站输入设置为 I0.1 ~ I0.7，利用软件监视 VB200 存储的数据即是 I0.1 ~ I0.7。

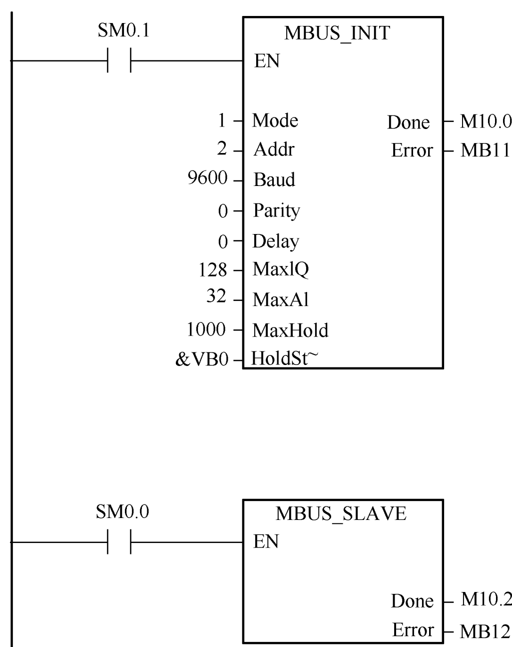


图 5.3-7　从站程序

实例 4　S7-200 PLC 的以太网通信

1. 控制要求

两台 S7-200 PLC 通过 CP243-1 扩展模块实现以太网的连接以及通过以太网对 PLC 进行编程与诊断。

2. 相关知识

工业以太网是基于国际标准 IEEE 802.3（Ethernet）的强大的区域和单元级设备联网的国际标准。在开放式 SIMATIC NET 通信系统中，工业以太网可以用作协调级和单元级网络。在技术上，工业以太网是一种基于屏蔽同轴电缆、双绞电缆而建立的电气网络，或是一种基于光纤电缆的光网络。

工业以太网模块 CP243-1 是一种通信处理器，SIMATIC PLC 通过以太网扩展模块 CP243-1 或者互联网扩展模块 CP243-1IT 来支持 TCP/IP 以太网通信，可以使用 STEP7-Micro/WIN 通过以太网对其 S7-200 进行远程组态、编程和诊断。要实现工业以太网与 S7-200 PLC 通信，必须使用 CP243-1（或 CP243-1IT）以太网模块，PC 上也要安装以太网网卡。CP243-1 可以与各种不同类型的 S7-200 CPU（222、224、226 和 226XM）相连接。但是每个 S7-200 CPU 只能连接一个 CP243-1，如果连接有多个 CP243-1，将不能保证 S7-200 系统的正常运行。

（1）工业以太网模块 CP243-1 技术规范（见表 5.4-1）。

表 5.4-1　CP243-1 技术规范

型号	6GK7 243-1EX00-0XE0
模块结构	S7-200 扩展模板
传输速率	10Mbit/s 和 100Mbit/s

（续）

闪存大小	1MB
SDRAM 存储大小	8MB
接口 连接到工业以太网 （10/100Mbit/s）	8 针 RJ45 插座
组态软件	STEP7-Micro/WIN（V3.2 SP1 和以上）
尺寸（W×H×D）	71.2mm×80mm×62mm
重量	约 150g
外部供电电压	DC24V（－15%～＋20%，DC20.4～28.8V）
功率消耗	1.75W
电流消耗	从 S7-200 总线供电：55mA 外部 DC 24V 供电：60mA
模块最大连接数量	最多 8 个 S7 连接（XPUT/XGET 和 READ/WRITE）＋1 个 STEP7 Micro/WIN 32 连接
允许环境条件： 工作温度 运输/贮存温度 相对湿度，最大 安装高度	水平安装，0～＋55℃ 垂直安装，0～＋45℃ －40～＋70℃ 95%，＋25℃时 海拔 2000m 以下，海拔越高，冷却越没有效果，需要降低最大工作温度
保护等级	IP20
以太网标准	IEEE 802.3
标准	CE 标记 UL 508 或 cULus CSA C22.2，Number 142 或 cULus FM 3611 EN 50081-2/EN 61000-6-4 EN 60529 EN 61000-6-2 EN 61131-2
启动时间或复位后的重新启动时间	约 10s
用户数据数量	作为客户机：对于 XPUT/XGET，212 个字节作为服务器：对于 XGET 或 READ，222 个字节；对于 XPUT 或 WRITE，212 个字节

（2）工业以太网模块 CP243-1 功能介绍

1）可通过 RJ45 进行以太网访问。

2）通过 S7-200 总线，即可与 S7-200 系统简单连接，可以实现一种灵活的分布式自动化架构。

3）通过工业以太网和 STEP7-Micro/WIN 32，实现 S7-200 系统的远程编程、组态和诊断。最多可以组态 8 个连接。

4）对于连接控制（保持活动状态），可以为主动和被动伙伴的所有的 TCP 传输连接组态。

5）可提供与 OPC 的连接。

6）CP243-1 允许 S7-200 编程软件 STEP7-Micro/WIN 通过工业以太网访问 S7-200。

7）无需重复进行编程/组态，即可更换模板（即插即用）。

8）S7 通信服务，“XPUT/XGET”，既可作为客户机，也可作为服务器，“READ/WRITE”作为服务器。

9）通过预设 MAC 地址（48 位数值），进行地址分配（在出厂时已对每个 CP243-1 进行了 MAC 地址分配。MAC 地址打印在附于上盖下面的标签上。使用 BOOTP 协议，通过预设的 MAC 地址，可以将 IP 地址分配给 CP243-1 通信处理器）。

3. 操作步骤

（1）硬件配置

实现本实例需要两台 S7-200 PLC 和两台 CP243-1 通信扩展模块，还有一台以太网交换机以及装有以太网卡的 PC 一台，三根普通网线和一根 PC/PPI 编程电缆，如图 5.4-1 所示。

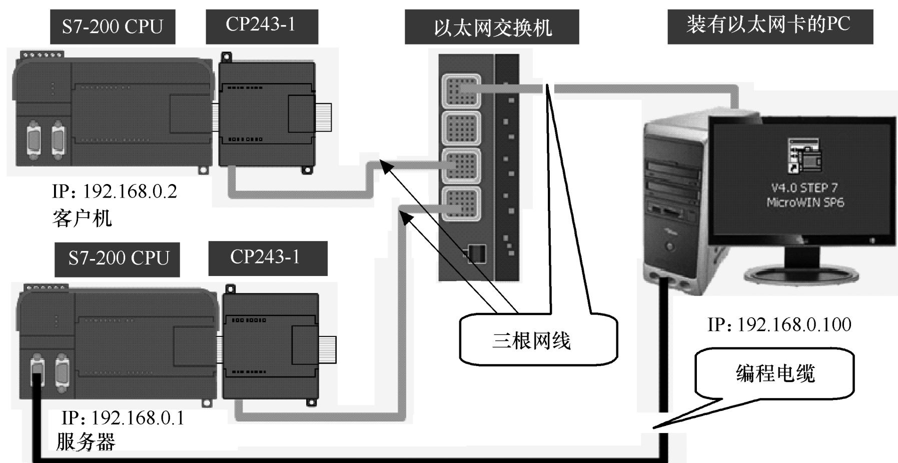


图 5.4-1　以太网通信的硬件配置图

（2）使用向导配置服务器

1）启动 STEP7-Micro/WIN 软件，找到“工具”菜单单击“以太网向导”命令出现以太网向导界面，如图 5.4-2 所示。

2）单击图 5.4-2 中的“下一步”按钮，进入模块配置界面，如图 5.4-3 所示，设置模块位置为“0”。如果在线情况下单击“读取模块”按钮，可以搜索在线的 CP243-1 模块，如图 5.4-4 所示。

3）单击图 5.4-4 中的“下一步”按钮，进入模块地址分配界面，如图 5.4-5 所示。在此模块设定 IP 地址为“192.168.0.1”，填写子掩码为“255.255.255.0”，模块连接类型为“自动检测通信”。

4）单击图 5.4-5 中的“下一步”按钮，进入模块命令字节和对等连接界面，如图 5.4-6 所示。Q 的地址使用系统默认设置，模块连接数目设置为“1”。



图 5.4-2　以太网向导配置 1

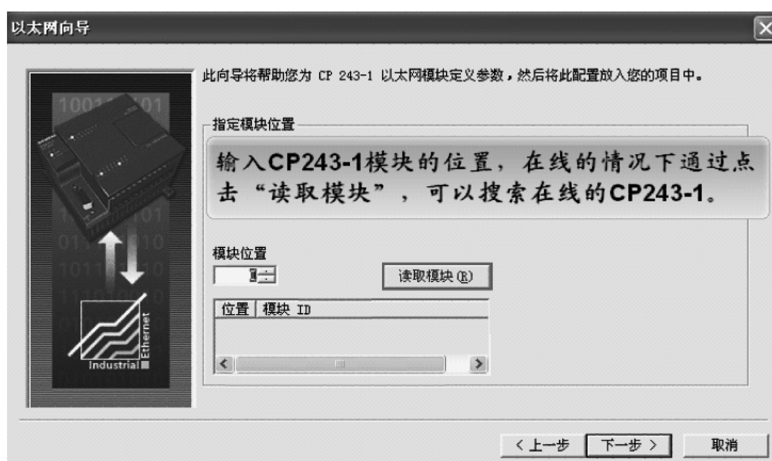


图 5.4-3　以太网向导配置 2

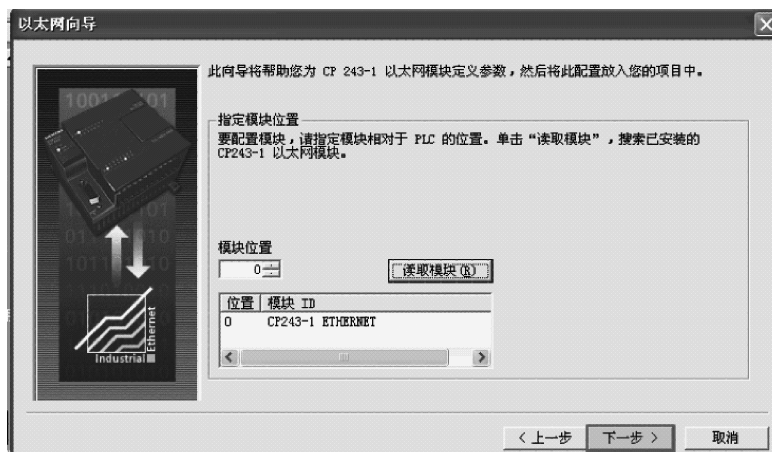


图 5.4-4　以太网向导配置 3

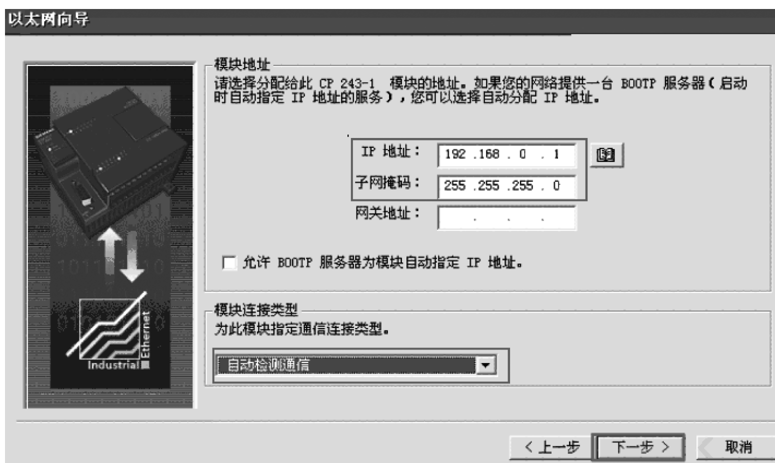


图 5.4-5　以太网向导配置 4

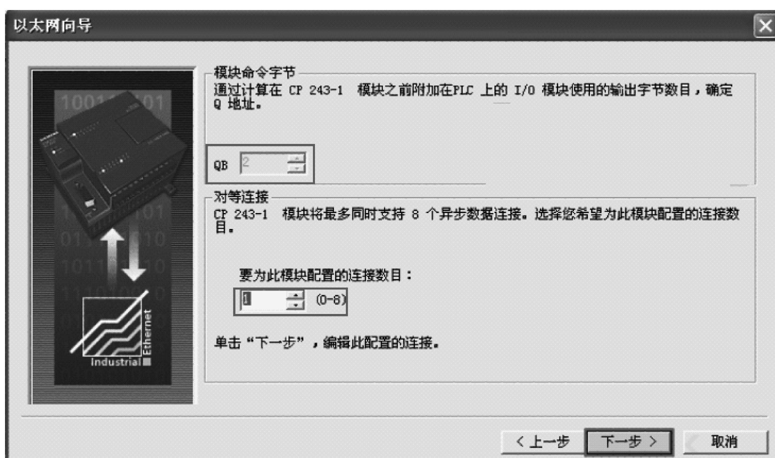


图 5.4-6　以太网向导配置 5

5）单击图 5.4-6 中的“下一步”按钮，出现“配置连接”对话框，如图 5.4-7 所示。选择连接为“服务器连接”，远程 TSAP 地址使用系统默认的设置“10”，本地 TSAP 地址自动生成，选中“接受所有连接请求”复选框，其他使用系统默认的设置。

6）单击图 5.4-7 中的“确认”按钮，进入 CRC 保护界面，如图 5.4-8 所示。选中“是，为数据模块中的此配置生成 CRC 保护”单选按钮，保持活动时间间隔为“30s”。

7）单击图 5.4-8 中的“下一步”按钮，出现为配置分配存储区界面，如图 5.4-9 所示，V 存储区的地址范围设置建议采用“建议地址”按钮自动分配。设置完成后，单击“下一步”按钮，进入图 5.4-10 所示的界面，按照提示单击“完成”按钮并保存设置，至此完成了服务器的以太网向导配置。

（3）使用向导配置客户机

1）启动软件进行客户机的配置，前两步同服务器的设置步骤。

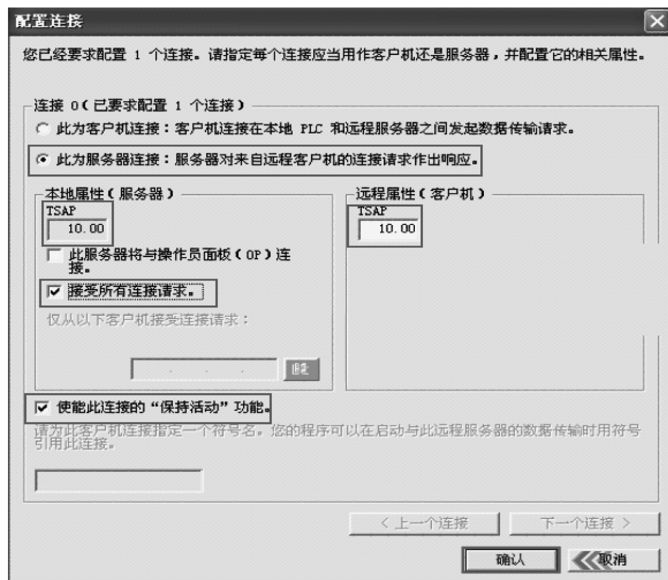


图 5.4-7　以太网向导配置 6

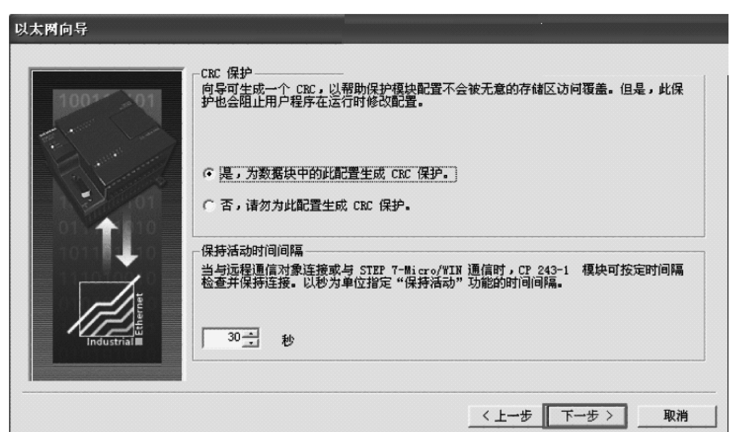


图 5.4-8　以太网向导配置 7

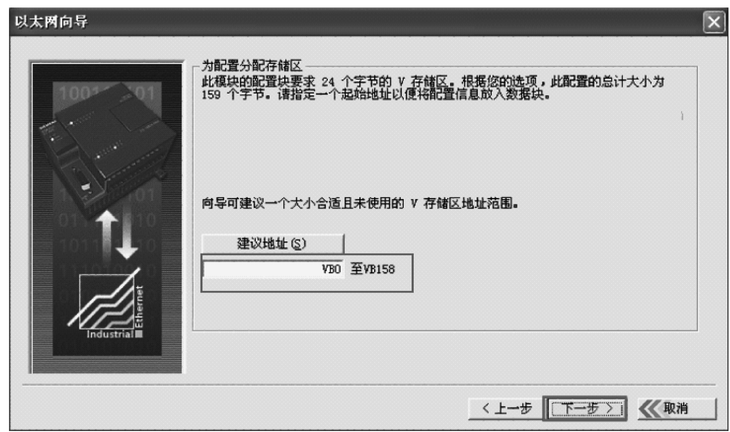


图 5.4-9　以太网向导配置 8

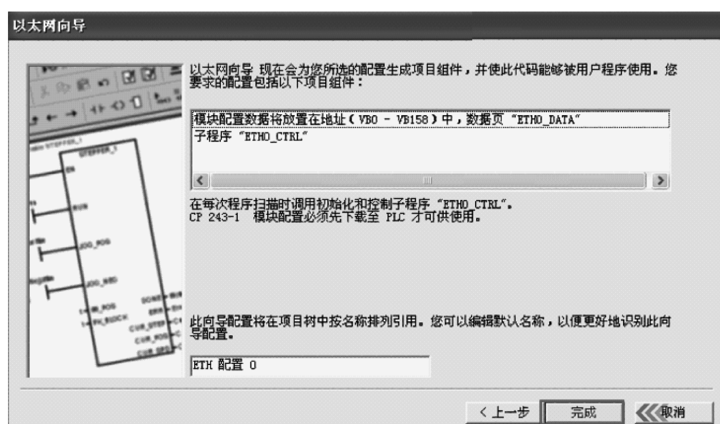


图 5.4-10　以太网向导配置 9

2）进入模块地址分配界面，如图 5.4-11 所示。在此模块设定 IP 地址为“192.168.0.2”，填写子掩码为“255.255.255.0”，模块连接类型为“自动检测通信”。

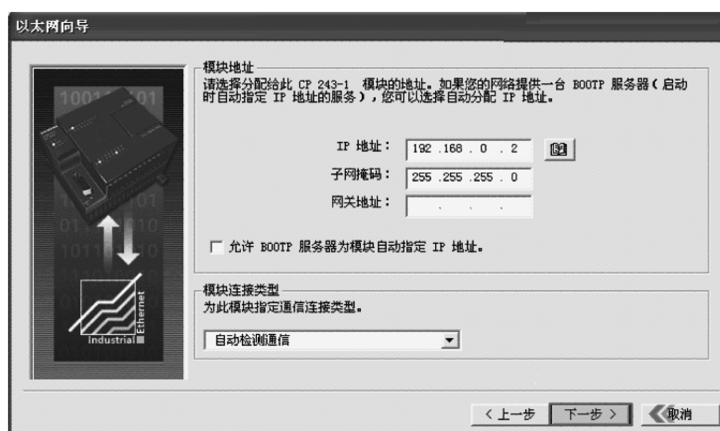


图 5.4-11　以太网向导客户机配置 1

3）单击图 5.4-11 中的“下一步”按钮，进入模块命令字节和对等连接界面，如图 5.4-12 所示。Q 的地址使用系统默认设置，模块连接数目设置为“1”。

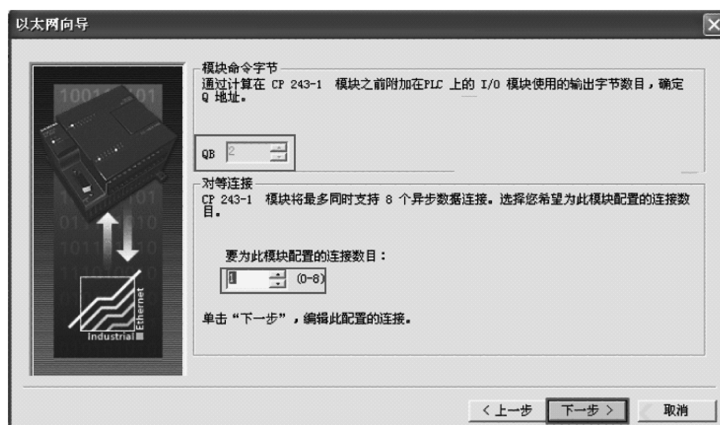


图 5.4-12　以太网向导客户机配置 2

4）单击图 5.4-12 中的“下一步”按钮，出现“配置连接”对话框，如图 5.4-13 所示，选择客户机连接，指定服务器 IP 地址为“192.168.0.1”。在此界面中，单击“数据传输”按钮，出现“配置 CPU 至 CPU 数据传输”对话框，如图 5.4-14 所示，单击“新传输”按钮，出现“添加一个新数据传输吗”对话框。

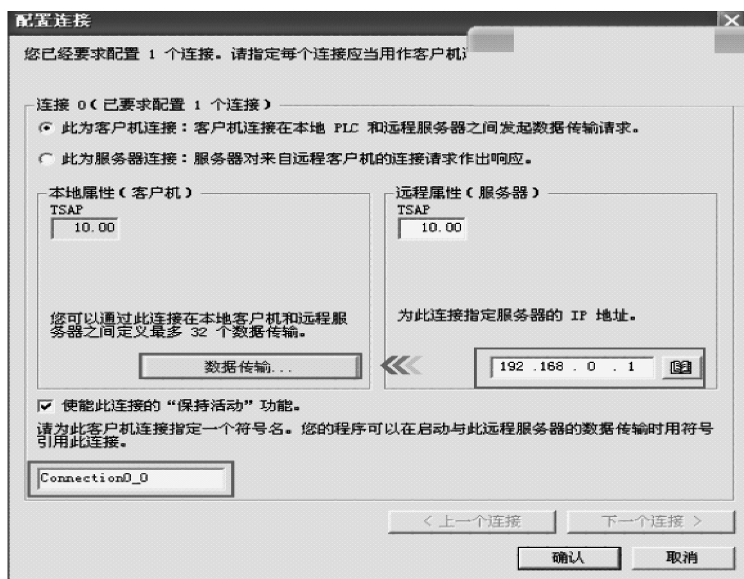


图 5.4-13　以太网向导客户机配置 3

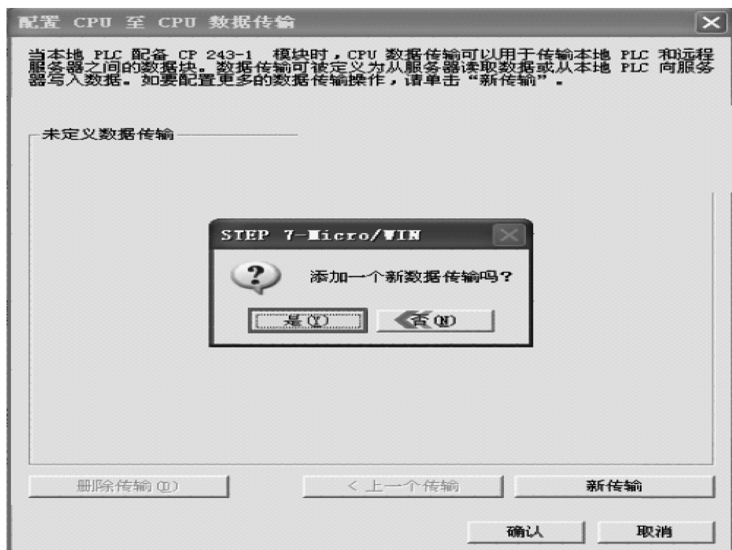


图 5.4-14　以太网向导客户机配置 4

5）单击图 5.4-14 中的“是”按钮，出现数据传输定义界面，如图 5.4-15 所示。数据传输选择“从远程服务器连接读取数据”，读取数据字节为“1”，选择从服务器的 IB0 读取数据，其他参数系统默认。

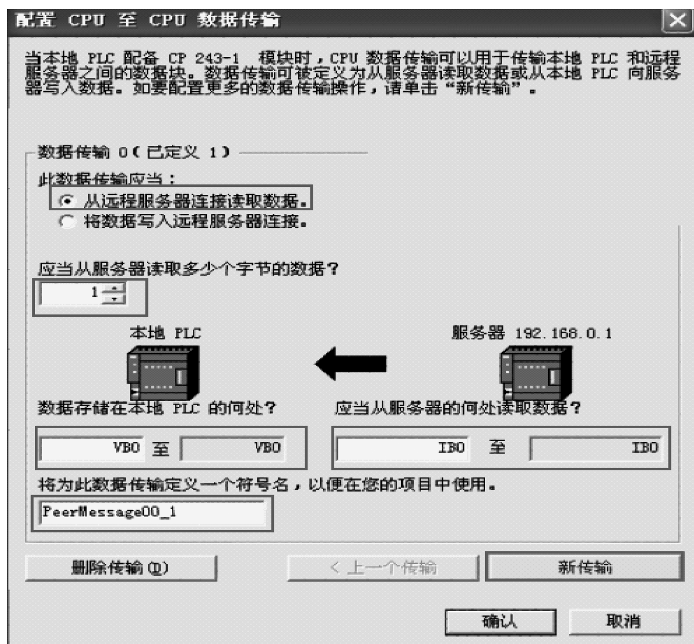


图 5.4-15　以太网向导客户机配置 5

6）单击图 5.4-15 中的“新传输”按钮，出现新的数据传输定义界面，如图 5.4-16 所示。数据传输选择“将数据写入远程服务器连接”，写入数据字节为“1”，本地 PLC 数据地址设为“VB1”，数据存入服务器的地址为“QB0”，其他参数系统默认。

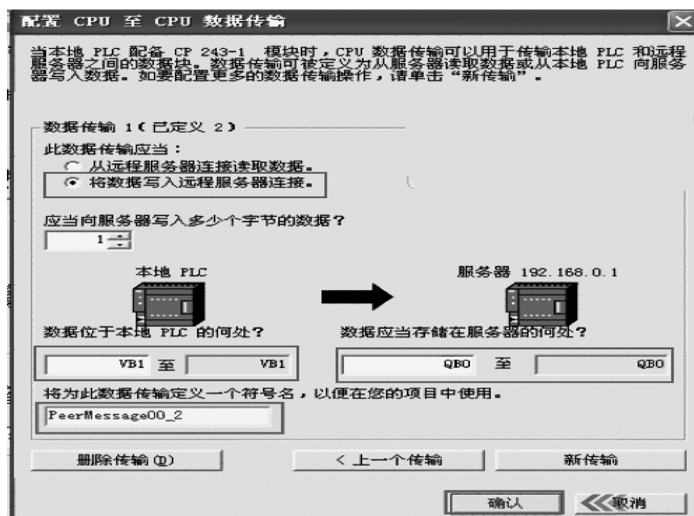


图 5.4-16　以太网向导客户机配置 6

7）单击图 5.4-16 中的“确认”按钮，出现如图 5.4-17 所示的界面，单击“确认”按钮，按照提示操作，出现如图 5.4-18 所示的配置存储区界面，单击“建议地址”按钮，系统自动分配地址。

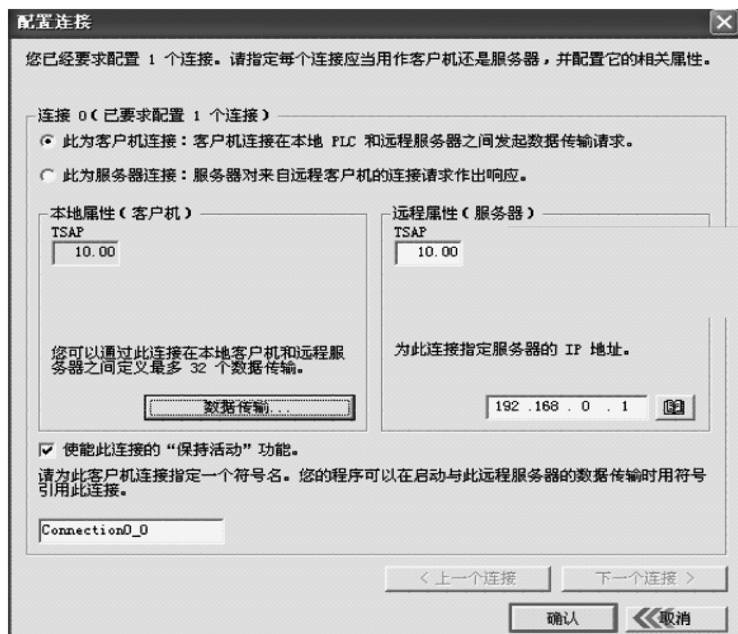


图 5. 4-17　以太网向导客户机配置 7

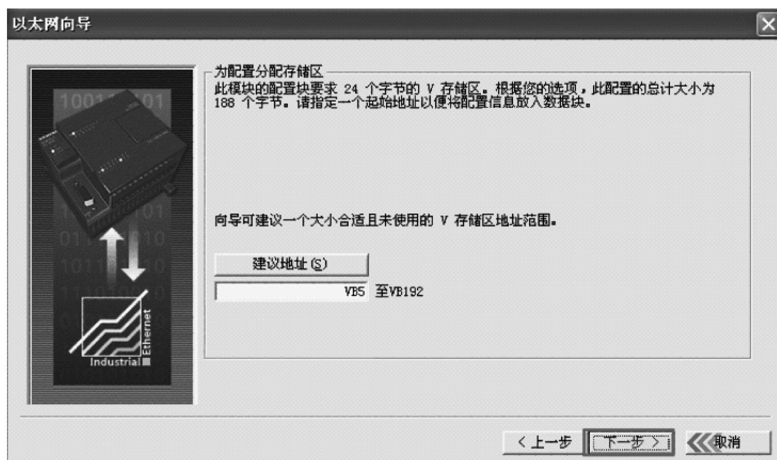


图 5. 4-18　以太网向导客户机配置 8

8）单击图 5. 4-18 中的“下一步”按钮，并按照提示操作，完成客户机的向导配置并保存该配置，如图 5. 4-19 所示。

（4）编写程序

1）服务器程序如图 5. 4-20 所示。利用向导组态完服务器后，调用向导生成的子程序编写服务器程序，编写完后下载到 S7-200 服务器中。

2）客户机程序如图 5. 4-21 所示。利用向导组态完客户机后，调用向导生成的子程序编写客户机程序，编写完后下载到 S7-200 客户机中。



图 5.4-19　以太网向导客户机配置 9

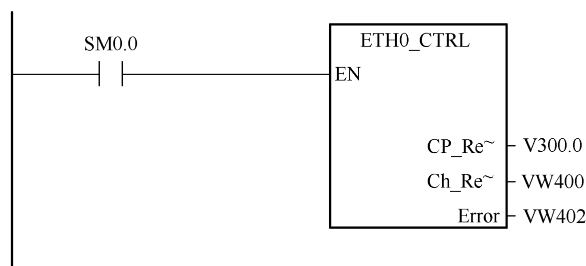


图 5.4-20　服务器程序

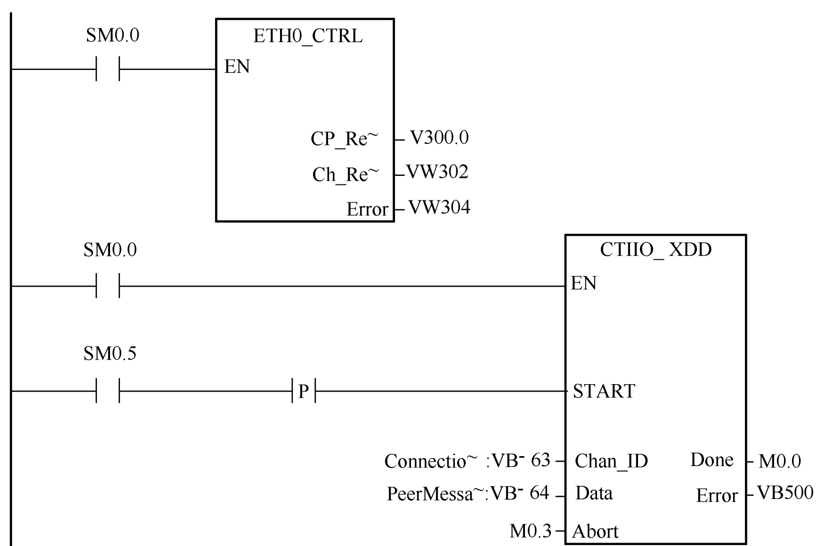


图 5.4-21　客户机程序

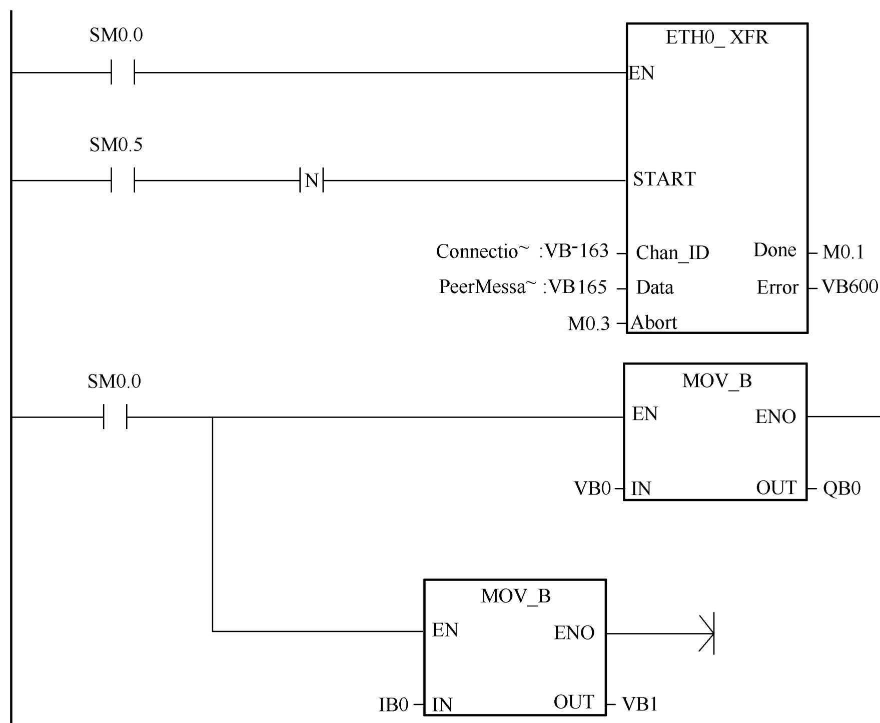


图 5.4-21　客户机程序（续）

（5）联机调试与诊断

两台 S7-200 都下载完程序后，均需要重新上电，并使其处于 RUN 状态下，此时对以太网的配置开始生效。分别拨动两台 PLC 上的 I0.0 ~ I0.7 输入端开关，观察 PLC 输出端 Q0.0 ~ Q0.7 的指示灯是否随着输入变化而变化。如果变化正常，说明通信成功，否则通信失败。

第六章

常用扩展模块

第一节　扩展模块介绍

当主机的 I/O 点数不够用或需要进行特殊功能的控制时，通常要进行 I/O 的扩展。I/O 扩展包括 I/O 点数的扩展和功能模块的扩展。不同的 CPU 有不同的扩展规范，其主要受 CPU 的寻址能力限制。

一、S7-200 CPU 数字量扩展模块

常用的数字量扩展模块有三类，即输入扩展模块、输出扩展模块、输入/输出扩展模块。S7-200 PLC 数字量扩展模块见表 6.1-1。

表 6.1-1　S7-200 PLC 数字量扩展模块

类　　型	型　　号	输入点数/类型	输出点数/类型
输入扩展模块	EM221	8 输入/DC 24V 光电隔离	
	EM221	8 输入/AC 120/230V	
输出扩展模块	EM222		8 输出/DC 24V 晶体管型
	EM222		8 输出/继电器型
	EM222		8 输出/AC 120/230V 晶闸管型
输入/输出扩展模块	EM223	4 输入/DC 24V 光电隔离	4 输出/DC 24V 晶体管型
	EM223	4 输入/DC 24V 光电隔离	4 输出/继电器型
	EM223	8 输入/DC 24V 光电隔离	8 输出/DC 24V 晶体管型
	EM223	8 输入/DC 24V 光电隔离	8 输出/继电器型
	EM223	16 输入/DC 24V 光电隔离	16 输出/DC 24V 晶体管型
	EM223	16 输入/DC 24V 光电隔离	16 输出/继电器型

二、S7-200 主机的模拟量扩展模块

当需要完成某些特殊功能的控制任务时，CPU 主机可以连接扩展模块，利用这些扩展模块进一步完善 CPU 的功能。常用的扩展模块有两类，即模拟量扩展模块、特殊功能模块。模拟量扩展模块有模拟量输入扩展模块、模拟量输出扩展模块、模拟量输入/输出扩展模块，其型号与用途见表 6.1-2。

表 6.1-2　模拟量扩展模块型号及用途

分　类	型　号	I/O 规格	功能及用途
模拟量输入扩展模块	EM231	AI4×12 位	4 路模拟输入，12 位 A-D 转换
		AI4×热电偶	4 路热电偶模拟输入
		AI4×RTD	4 路热电阻模拟输入
模拟量输出扩展模块	EM232	AQ2×12 位	2 路模拟输出
模拟量输入/输出扩展模块	EM235	AI4/AQ1×12	4 路模拟输入，1 路模拟输出，12 位转换

1. 模拟量输入模块

模拟量输入在过程控制中的应用很广，如常用的温度、压力、速度、流量、酸碱度、位移的各种工业检测都是对应于电压、电流的模拟量值，再通过一定运算（PID）后，控制生产过程达到一定的目的。模拟量输入电平大多是从传感器通过变换后得到的，模拟量的输入信号为 4～20mA 的电流信号或 1～5V、－10～10V、0～10V 的直流电压信号。而 PLC 只能接收数字量信号，为实现模拟量控制，必须先对模拟量进行模-数（A-D）转换，将模拟信号转换成 PLC 所能接收的数字信号。模拟量输入单元一般由滤波、A-D 转换器、光耦合器隔离等组成，其原理框图如图 6.1-1 所示。

图 6.1-2 所示为 EM231 模拟量输入模块，其主要由滤波、A-D 转换器、光耦合器隔离、内部电路组成。当输入信号通过滤波、运算放大器的放大和量程变换后，转换成 A-D 转换器能够接收的电压范围，经过 A-D 转换器后的数字量信号，再经光耦合器隔离后进入 PLC 的内部电路。根据 A-D 转换的分辨率不同，模拟量输入单元能提供 8 位、10 位、12 位或 16 位等精度的各种位数的数字量信号并传送给 PLC 以进行处理。

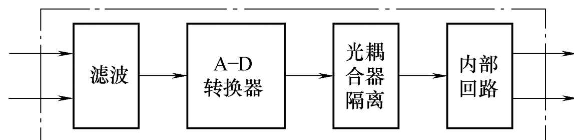


图 6.1-1　模拟量输入单元原理框图

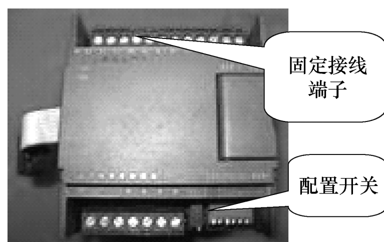


图 6.1-2　EM231 模拟量输入模块

2. 模拟量输出模块

模拟量输出模块是将中央处理器的二进制数字信号转换成 4～20mA 的电流输出信号或 0～10V、1～5V 的电压输出信号，提供给执行机构，以满足生产过程现场连续信号的控制要求。模拟量输出单元一般由光耦合器隔离、D-A 转换器和信号驱动等组成，其原理框图如图 6.1-3 所示。图 6.1-4 所示为 EM232 模拟量输出模块，其一般具有 2 路或 4 路模拟量输出通道。

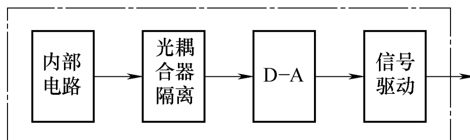


图 6.1-3　模拟量输出单元原理框图

3. 模拟量输入/输出模块

图 6.1-5 所示为 EM235 模拟量输入/输出模块，其具有 4 路模拟量输入通道、1 路模拟

量输出通道。



图 6.1-4　EM232 模拟量输出模块



图 6.1-5　EM235 模拟量输入/输出模块

该模块的模拟量输入功能同 EM231 模拟量输入模块，技术参数也基本相同。电压输入范围有所不同，单极性为 0 ~ 10V、0 ~ 5V、0 ~ 1V、0 ~ 500mV、0 ~ 100mV、0 ~ 50mV；双极性为 ±10V、±5V、±2.5V、±1V、±500mV、±250mV、±100mV、±50mV、±25mV。

该模块的模拟量输出功能同 EM232 模拟量输出模块，技术参数也基本相同。

4. 模拟量模块的接线

1）CPU224XP 本身集成模拟量接线。CPU224XP 本身集成有 2 路电压输入和 1 路模拟量输出，接线方法如图 6.1-6 所示。

2 路电压输入接线分别为 A + 和 M、B + 和 M，此时只能输入 ±10V 电压信号。1 路模拟量输出信号，如果是电流输出将负载接在 I 和 M 端子之间，如果是电压输出将负载接在 V 和 M 端子之间。

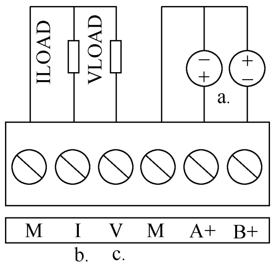


图 6.1-6　CPU224XP 集成模拟量接线图

2）模拟量输入模块 EM231、模拟量输出模块 EM232 和模拟量输入/输出模块 EM235 的应用接线图如图 6.1-7 所示。

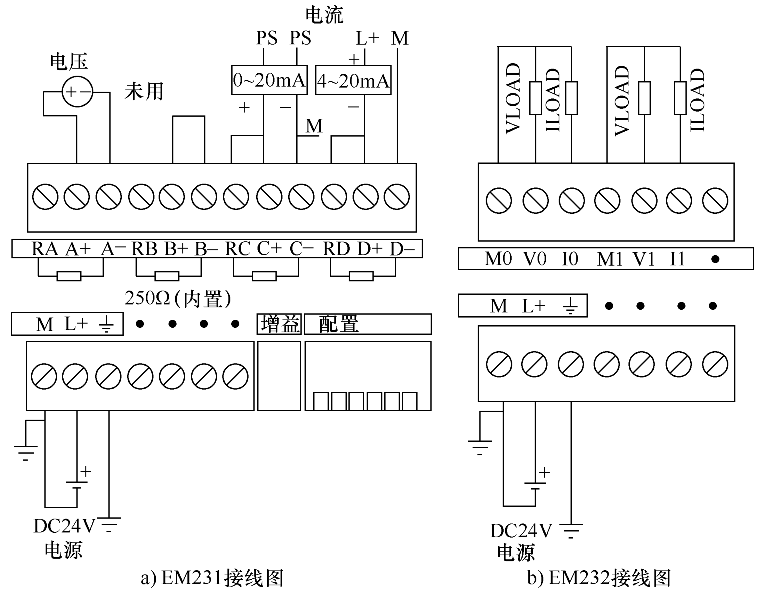
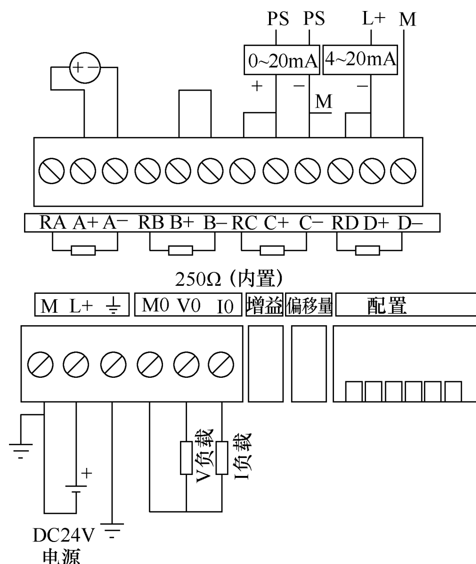


图 6.1-7　接线图



c) EM235接线图

图 6.1-7　接线图（续）

三、特殊功能模块

S7-200 主机的特殊功能模块有多种类型，如 EM253 位置控制模块、EM277 Profibus-DP 模块、EM241 调制解调器模块、CP243-1 以太网模块、CP243-2 AS-I 接口模块等，如图 6.1-8 所示。


a)EM253模块　b)EM241模块　c)CP243-2 模块　d)CP243-1模块

图 6.1-8　特殊功能模块实物图

第二节　扩展模块的应用

一、I/O 点数扩展和编址

S7-200 CPU22X 系列的每种主机所提供的本机 I/O 点的 I/O 地址都是固定的，进行扩展时，可以在 CPU 右边连接多个扩展模块，每个扩展模块的组态地址编号均取决于各模块的类型和该模块在 I/O 链中所处的位置。输入与输出模块的地址不会冲突，模拟量控制模块地

址也不会影响数字量。编址方法与原则如下：

1）同类型输入或输出的模块按顺序进行编址。

2）数字量模块总是保留以 8 位（1 个字节）递增的过程映像寄存器空间。如果模块没有给保留字节中每一位提供相应的物理点，那些未用位不能分配给 I/O 链中的后续模块。对于输入模块，这些保留字节中未使用的位会在每个输入刷新周期中被清零。

3）模拟量 I/O 点总是以两点递增的方式来分配空间。如果 CPU 或模块在为物理 I/O 点分配地址时未用完 1 个字节，那些未用的位也不能分配给 I/O 链中的后续模块。

例如，某一控制系统选用 CPU224，系统所需的输入/输出点数为数字量输入 24 点、数字量输出 20 点、模拟量输入 6 点和模拟量输出 2 点。

本系统可有多种不同模块的选取组合，并且各模块在 I/O 链中的位置排列方式也可能有多种，图 6.2-1 所示为其中的一种模块连接形式，表 6.2-1 所示为其对应的各模块的编址情况。

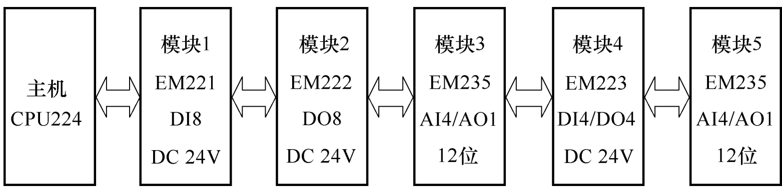


图 6.2-1　模块连接形式

表 6.2-1　各模块的编址

主机 I/O	模块 1 I/O	模块 2 I/O	模块 3 I/O	模块 4 I/O	模块 5 I/O
I0.0　Q0.0	I2.0	Q2.0	AIW0　AQW0	I3.0　Q3.0	AIW8　AQW4
I0.1　Q0.1	I2.1	Q2.1	AIW2	I3.1　Q3.1	AIW10
I0.2　Q0.2	I2.2	Q2.2	AIW4	I3.2　Q3.2	AIW12
I0.3　Q0.3	I2.3	Q2.3	AIW6	I3.3　Q3.3	AIW14
I0.4　Q0.4	I2.4	Q2.4			
I0.5　Q0.5	I2.5	Q2.5			
I0.6　Q0.6	I2.6	Q2.6			
I0.7　Q0.7	I2.7	Q2.7			
I1.0　Q1.0					
I1.1　Q1.1					
I1.2					
I1.3					
I1.4					
I1.5					

二、模拟量扩展模块的应用

1. 模拟量输入/输出映像寄存器

S7-200 的模拟量输入电路是将外部输入的模拟量信号（电流或电压）转换成 1 个字长（16 位）的数字量（0 ~ 32 000）存入模拟量输入映像寄存器区域，可以用区域标识符

(AI)、数据长度（W）和模拟通道的起始地址读取这些量，其格式为 AIW［起始字节地址］。因为模拟输入量为 1 个字长（16 位），即 2 个字节，且从偶数字节开始存放，所以必须从偶数字节地址读取这些值，如 AIW0、AIW2、AIW4 等。模拟量输入值为只读数据。

S7-200 CPU 的模拟量输出电路是将模拟量输出映像寄存器区域的 1 个字长（16 位）的数字量（0 ~ 32 000）转换为模拟量信号（电流或电压）输出，可用区域标识符（AQ）、数据长度（W）和模拟通道的起始地址存储这些量，其格式为 AQW［起始字节地址］。因为模拟输出量为 1 个字长（16 位），即 2 个字节，且从偶数字节开始存放，如 AQW0、AQW2、AQW4 等。模拟量输出值是只写数据。

对模拟量输入/输出是以 2 个字（W）为单位分配地址，每路模拟量输入/输出占用 1 个字（2 个字节）。如果有 3 路模拟量输入，需分配 4 个字（AIW0、AIW2、AIW4、AIW6），其中没有被使用的字 AIW6，不可被占用或分配给后续模块。如果有 1 路模拟量输出，需分配 2 个字（AQW0、AQW2），其中没有被使用的字 AQW2，不可被占用或分配给后续模块。

模拟量输入/输出的地址编号范围根据 CPU 的型号不同而有所不同，CPU222 为 AIW0 ~ AIW30/AQW0 ~ AQW30，CPU224/226 为 AIW0 ~ AIW62/AQW0 ~ AQW62。

2. PLC 模拟量扩展模块的应用

在工业控制中，某些输入量（如压力、温度、流量、转速等）是模拟量，某些执行机构（如电动调节阀、变频器等）要求 PLC 输出模拟信号。模拟量首先被传感器和变送器转换为标准量程的电流或电压，如直流 4 ~ 20mA、1 ~ 5V 或 0 ~ 10V 等。PLC 用 A-D 转换器将它们转换成数字量。带正负号的电流或电压在 A-D 转换后用二进制补码表示。D-A 转换器将 PLC 的数字输出量转换为模拟电压或电流，再去控制执行机构。模拟量 I/O 模块的主要任务就是实现 A-D 转换（模拟量输入）和 D-A 转换（模拟量输出），如图 6.2-2 所示。

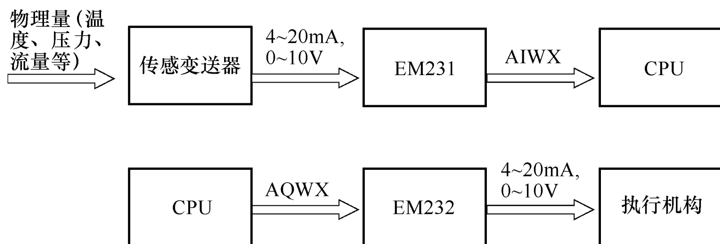


图 6.2-2　工程量与模拟量、数字量转化

（1）模拟量输入模块 EM231 的应用

1）EM231 模块接线。通过 A-D 模块，S7-200 CPU 可以将外部的模拟量（电流或电压）转换成 1 个字长（16 位）的数字量（0 ~ 32 000）。

图 6.2-3 为 EM231 的端子接线及 DIP 开关示意图。

2）EM231 模块的配置和校准。使用 EM231 和 EM235 输入模拟量时，首先要进行模块的配置和校准。通过调整模块中的 DIP 开关，可以设定输入模拟量的种类（电流、电压）以及模拟量的输入范围、极性，见表 6.2-2。

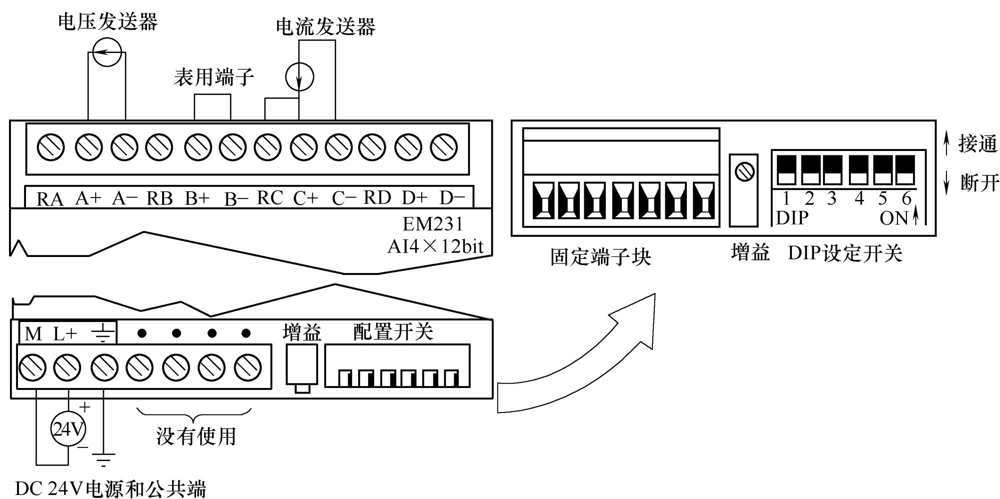


图 6.2-3　EM231 的端子接线及 DIP 开关示意图

表 6.2-2　EM231 选择模拟量输入范围的开关表

<table>
<tr><td colspan="3">单极性</td><td rowspan="2">满量程输入</td><td rowspan="2">分辨率</td><td colspan="3" rowspan="2">双极性</td><td rowspan="3">满量程输入</td><td rowspan="3">分辨率</td></tr>
<tr><td>SW1</td><td>SW2</td><td>SW3</td></tr>
<tr><td rowspan="3">ON</td><td>OFF</td><td>ON</td><td>0 ~ 10V</td><td>2.5mV</td><td>SW1</td><td>SW2</td><td>SW3</td></tr>
<tr><td rowspan="2">ON</td><td rowspan="2">OFF</td><td>0 ~ 5V</td><td>1.25mV</td><td rowspan="2">OFF</td><td>OFF</td><td>ON</td><td>±5V</td><td>2.5mV</td></tr>
<tr><td>0 ~ 20mA</td><td>5μA</td><td>ON</td><td>OFF</td><td>±2.5V</td><td>1.25mV</td></tr>
</table>

注：双极性信号就是信号在变化的过程中要经过“零”，单极性不过“零”。由于模拟量转换为数字量是有符号整数，所以双极性信号对应的数值会有负数。在 S7-200 中，单极性模拟量输入/输出信号的数值范围是 0 ~ 32 000，双极性模拟量信号的数值范围是 −32 000 ~ +32 000。

设定模拟量输入类型后，需要进行模块的校准，此操作需通过调整模块中的“增益调整”电位器来实现。校准调节影响所有的输入通道。即使在校准以后，如果模拟量多路转换器之前的输入电路元件值发生变化，从不同通道读入同一个输入信号，其信号值也会有微小的不同。校准输入的步骤如下：

① 切断模块电源，用 DIP 开关选择需要的输入范围；

② 接通 CPU 和模块电源，使模块稳定 15min；

③ 用一个变送器、一个电压源或电流源，将零值信号加到模块的一个输入端；

④ 读取该输入通道在 CPU 中的测量值；

⑤ 调节模块上的 OFFSET（偏置）电位器，直到读数为零或需要的数字值；

⑥ 将一个工程量的最大值（或满刻度模拟量信号）接到某一个输入端子，调节模块上的 GAIN（增益）电位器，直到读数为 32 000 或需要的数字值。

⑦ 必要时重复上述校准偏置和增益的过程。

如输入电压范围是 0 ~ 10V 的模拟量信号，则对应的数字量结果应为 0 ~ 32 000；电压为 0V 时，数字量不一定是 0，可能有一个偏置值，如图 6.2-4 所示。

3）输入模拟量的读取。每个模拟量占用 1 个字长（16 位），其中数据占 12 位。依据输入模拟量的极性，数据格式有所不同，如图 6.2-5 所示。

对于单极性数据格式（0～10V、0～5V），其最大值为 $2^{15}-2^{3}=32\ 760$，差值为 32 760－32 000＝760，可以通过调偏差/增益系统完成。

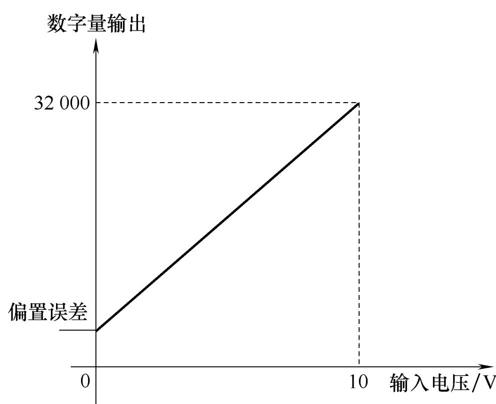


图 6. 2-4　模拟量输入与数字量输出关系

模拟量转换为数字量的 12 位读数是左对齐的。对单极性格式，最高位为符号位，最低 3 位是精度位，即 A-D 转换是以 8 为单位进行的；对双极性格式，最低 4 位为精度位，即 A-D 转换是以 16 为单位进行的。

模拟量输入模块有两个参数容易混淆，即模拟量转换的分辨率和模拟量转换的精度（误差）。分辨率是 A-D 转换芯片的转换精度，即用多少位的数值来表示模拟量。若 S7-200 模拟量模块的转换分辨率是 12 位，能够反映模拟量变化的最小单位是满量程的 $1/2^{12}$ 即 1/4096。模拟量转换的精度除了取决于 A-D 转换的分辨率，还受到转换芯片外围电路的影响。在实际应用中，输入的模拟量信号会有波动、噪声和干扰，内部模拟电路也会产生噪声、漂移，这些都会对转换的最后精度造成影响。这些因素造成的误差要大于 A-D 芯片的转换误差。



MSB　　　　LSB

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	12位数据												0	0	0

a)单极性格式

MSB　　　　LSB

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
12位数据												0	0	0	0

b)双极性格式

图 6. 2-5　模拟量输入数据格式

在读取模拟量时，利用数据传送指令 MOV_W，可以从指定的模拟量输入通道将其读取到内存中，然后根据极性，利用移位指令或整数除法指令将其规格化，以便于处理数据值部分。

（2）模拟量输出模块 EM232 的应用

1）EM232 模块接线。通过 D-A 模块，S7-200 CPU 把 1 个字长（16 位）的数字量（0～32 000）按比例转换成电流或电压。图 6. 2-6 所示为模拟量输出 EM232 端子接线及内部结构。

2）模拟量的输出。模拟量的输出范围为－10～＋10V 和 0～20mA（由接线方式决定），对应的数字量分别为－32 000～＋32 000 和 0～32 000。

图 6. 2-7 所示模拟量数据输出值是左对齐的。对于单极性格式，最高有效位是符号位，0 表示正值；对于双极性格式，最低 4 位是 4 个连续的 0，在转换为模拟量输出值时将自动屏蔽，而不会影响输出信号值。

（3）模拟量数据的处理

1）模拟量输入信号的整定。通过模拟量输入模块转换后的数字信号直接存储在 S7-200 PLC 的模拟量输入存储器 AIW 中。这种数字量与被转换的结果之间有一定的函数对应关系，

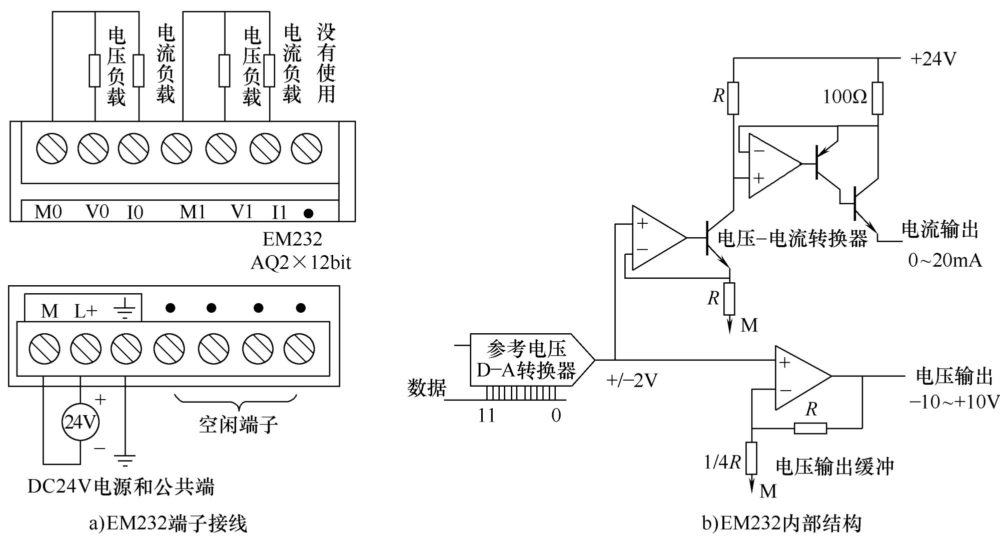


图 6. 2-6　模拟量输出 EM232 端子接线及内部结构



图 6. 2-7　模拟量数据输出

但在数值上并不相等，必须经过某种转换才能使用。这种将模拟量输入模块转换后的数字信号在 PLC 内部按一定函数关系进行转换的过程称为模拟量输入信号的整定。

模拟量输入信号的整定通常需要考虑以下几个问题。

① 模拟量输入值的数字量表示方法。模拟量输入值的数字量表示方法即模拟量输入模块数据的位数是多少，是否从数据字的第 0 位开始。若不是，应进行移位操作使数据的最低位排列在数据字的第 0 位上，以保证数据的准确性。如 EM231 模拟量输入模块，在单极性信号输入时，模拟量的数据值是从第 3 位开始的，因此数据整定的任务是把该数据字右移 3 位。

② 模拟量输入值的数字量表示范围。该范围由模拟量输入模块的转换精度决定。如果输入量的范围大于模块可能表示的范围，则可以使输入量的范围限定在模块表示的范围内。

③ 系统偏移量的消除。系统偏移量是指在无模拟量信号输入情况下由测量元件的测量误差及模拟量输入模块的转换死区所引起的，具有一定数值的转换结果。消除这一偏移量的方法是在硬件方面进行调整（如调整 EM231 中偏置电位器）或使用 PLC 的运算指令消除。

④ 过程量的最大变化范围。过程量的最大变化范围与转换后的数字量最大变化范围应有一一对应的关系，这样就可以使转换后的数字量精确地反映过程量的变化。如用 0 ~ 0FH 反映 0 ~ 10V 的电压与用 0 ~ FFH 反映 0 ~ 10V 的电压相比较，后者的灵敏度或精确度显然要

比前者高得多。

⑤ 标准化问题。从模拟量输入模块采集到的过程量都是实际的工程量，其幅度、范围和测量单位都不同，在 PLC 内部进行数据运算之前，必须将这些值转换为无量纲的标准格式。

⑥ 数字量滤波问题。电压、电流等模拟量常常会因为现场干扰而产生较大波动，这种波动经 A-D 转换后亦反映在 PLC 的数字量输入端。若仅用瞬时采样值进行控制计算，将会产生较大误差，因此有必要进行滤波。

工程上的数字滤波方法有平均值滤波、去极值平均滤波以及惯性滤波法等。

2）模拟量输出信号的整定。在 PLC 内部进行模拟量输入信号处理时，通常把模拟量输入模块转换后的数字量转换为标准工程量，经过工程实际需要的运算处理后，可得出上下限报警信号及控制信息。报警信息经过逻辑控制程序可直接通过 PLC 的数字量输出点输出，而控制信息需要暂存到模拟量存储器 AQWX 中，经模拟量输出模块转换为连续的电压或电流信号输出到控制系统的执行部件，以便进行调节。模拟量输出信号的整定就是要将 PLC 的运算结果按照一定的函数关系转换为模拟量输出寄存器中的数字值，以备模拟量输出模块转换为现场需要的输出电压或电流。

已知在某温度控制系统中由 PLC 控制温度的升降。当 PLC 的模拟量输出模块输出 10V 电压时，要求系统温度达到 500℃，现 PLC 的运算结果为 200℃，则应向模拟量输出存储器 AQWX 写入的数字量为多少？这就是一个模拟量输出信号的整定问题。

显然，解决这一问题的关键是要了解模拟量输出模块中的数字量与模拟量之间的对应关系，这一关系通常为线性关系。如 EM232 模拟量输出模块输出的 0 ~ 10V 电压信号对应的内部数字量为 0 ~ 32 000。上述运算结果 200℃所对应的数字量可用简单的算术运算程序得出。

例如，某管道水的压力是 0 ~ 1MPa，通过变送器转化成 4 ~ 20mA 输出，经过 EM231 的 A-D 转换，0 ~ 20mA 对应的数字量范围是 0 ~ 32 000，当压力大于 0.8MPa 时指示灯亮。

工程量与模拟量、模拟量与数字量的对应关系如图 6.2-8 所示。

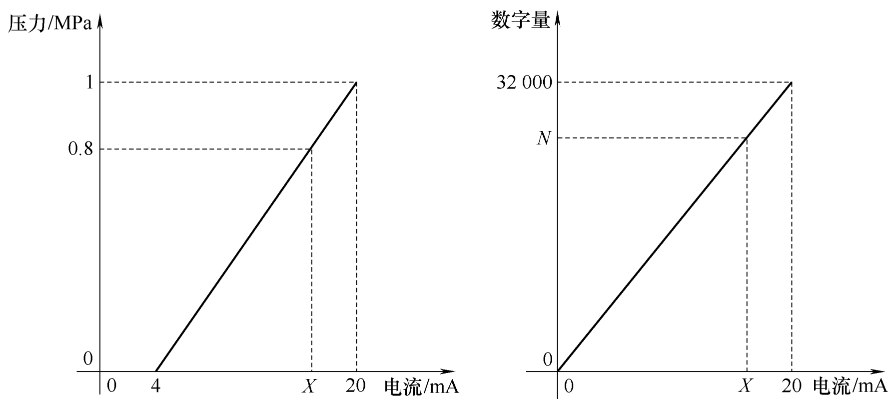


图 6.2-8　工程量与模拟量、模拟量与数字量的对应关系

0.8MPa 时的电流值为 $X = [(20-4)\times(0.8-0)/(1-0)]+4$；

0.8MPa 时的信号量是 $X = 16.8\text{mA}$；

对应的数字量是 $N = [(32\ 000-0)\times(16.8-0)/(20-0)]+0$；

0.8MPa 时的数字量是 $N=26\,880$。

PLC 程序如图 6.2-9 所示。

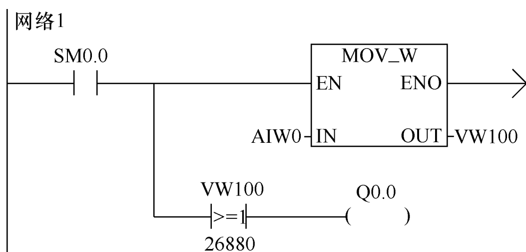


图 6.2-9　PLC 程序

（4）使用模拟量模块时的注意事项

1）模拟量模块有专用的扁平电缆与 CPU 通信，并通过此电缆由 CPU 向模拟量模块提供 DC 5V 的电源。此外，模拟量模块必须外接 DC 24V 电源。

2）每个模拟量模块能同时输入/输出电流或者电压信号。当模拟量模块的输入点/输出点有信号输入或者输出时，LED 指示灯不会亮，这点与数字量模块不同，因为西门子模拟量模块上的指示灯没有与电路相连。

3）一般电压信号比电流信号容易受干扰，应优先选用电流信号。电压型的模拟量信号，由于输入端的内阻很高（S7-200 的模拟量模块为 10MΩ），极易引入干扰。一般电压信号用于控制设备柜内的电位器设置，或者距离非常近、电磁环境好的场合。电流型信号不容易受到传输线沿途的电磁干扰，因而在工业现场获得广泛的应用。电流信号可以传输比电压信号远得多的距离。

4）对于模拟量输出模块，电压型和电流型输出信号的接线不同，各自的负载接到各自的端子上。

5）模拟量输出模块总是要占据两个通道的输出地址。即便有些模块（EM235）只有一个实际输出通道，也要占用两个通道的地址。在编程计算机和 CPU 实际联机时，使用 Micro/WIN 的菜单命令“PLC”→“信息（Information）”，可以查看 CPU 和扩展模块的实际 I/O 地址分配。

第七章

常用机床电气的 PLC 改造实例

实例 1　CA6140 卧式车床的 PLC 控制

1. CA6140 车床的控制要求

车床是一种应用极为广泛的金属切削机床，能够车削外圆、内圆、端面、螺纹以及切断及割槽等，并可以装上钻头或铰刀进行钻孔和铰孔等加工。图 7.1-1 所示为机械加工中应用较为广泛的 CA6140 型卧式车床，它主要由床身、主轴箱、进给箱、溜板箱、刀架、卡盘、尾座、丝杠和光杠等部分组成。

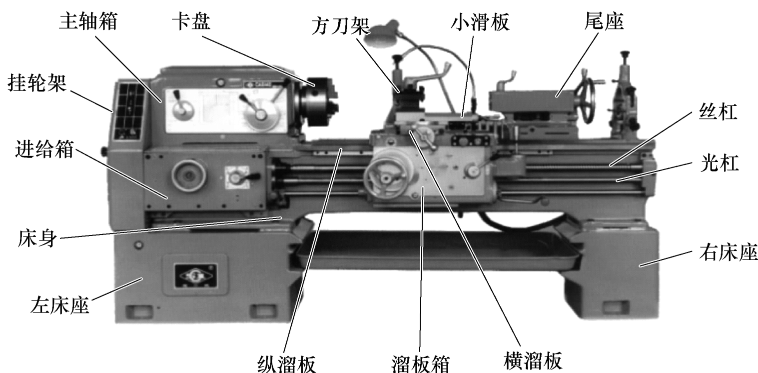


图 7.1-1　CA6140 型卧式车床

图 7.1-2 为 CA6140 车床的电路图，车床共有三台电动机。

主轴电动机 M1：带动主轴旋转和刀架做进给运动，由交流接触器 KM 控制，热继电器 FR1 作过载保护，断路器 QF 作短路保护。

冷却泵电动机 M2：输送切削液，由中间继电器 KA1 控制，热继电器 FR2 作过载保护，FU1 作短路保护。

刀架快速移动电动机 M3：拖动刀架的快速移动，由中间继电器 KA2 控制。由于刀架移动是短时工作，用点动控制，未设过载保护，FU1 兼作短路保护。

CA6140 车床辅助控制有信号灯、照明灯。

2. CA6140 车床电气控制线路分析

（1）主电路分析

CA6140 卧式车床的电源由钥匙开关 SB 控制，将 SB 向右旋转，再扳动断路器 QF 将三相电源引入。电气控制线路中共有三台电动机：M1 为主轴电动机，带动主轴旋转和刀架做

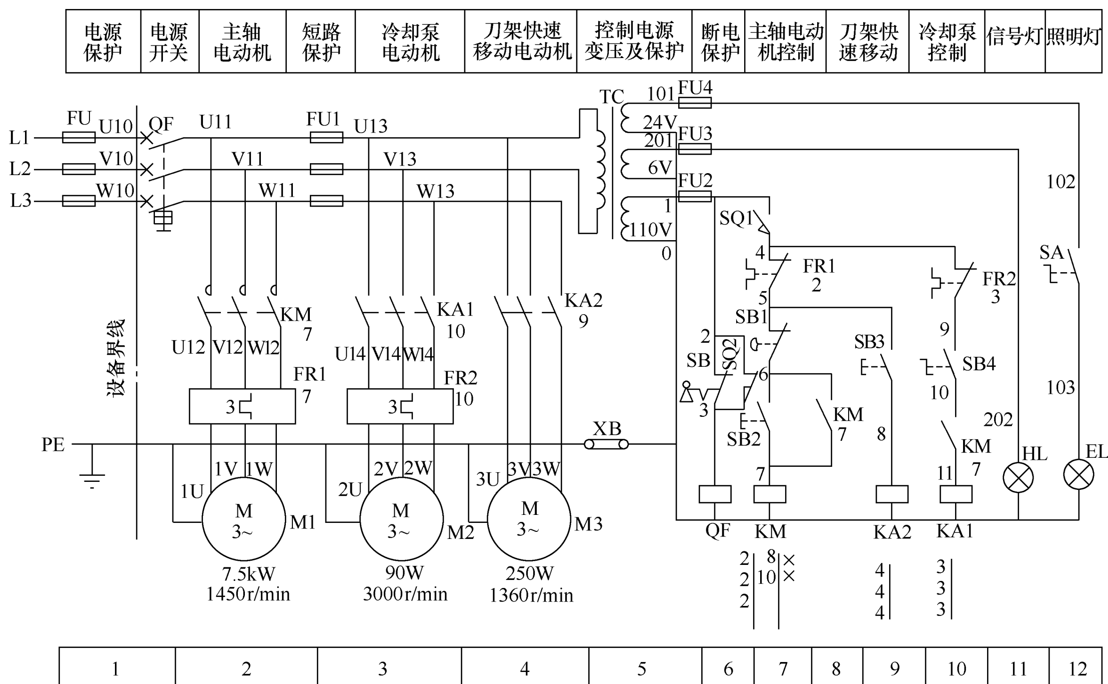


图 7.1-2　CA6140 车床的电路图

进给运动；M2 为冷却泵电动机，用以输送冷却液；M3 为刀架快速移动电动机，用以拖动刀架快速移动。主电路的控制和保护电器见表 7.1-1。

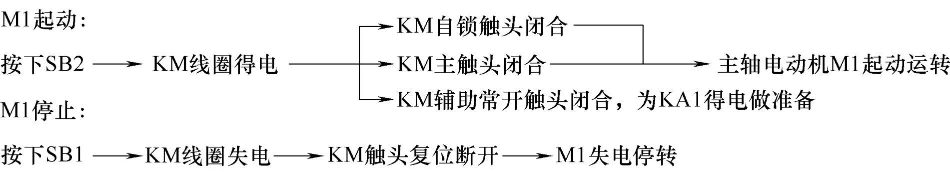
表 7.1-1　主电路的控制和保护电器

名称及代号	作　　用	控 制 电 器	过载保护电器	短路保护电器
主轴电动机 M1	带动主轴旋转和刀架做进给运动	接触器 KM	热继电器 FR1	低压断路器 QF
冷却泵电动机 M2	供应冷却液	中间继电器 KA1	热继电器 FR2	熔断器 FU1
快速移动电动机 M3	拖动刀架快速移动	中间继电器 KA2	无	熔断器 FU1

（2）控制电路分析

控制电路通过控制变压器 TC 输出的 110V 交流电压供电，由熔断器 FU2 做短路保护。在正常工作时，行程开关 SQ1 的常开触头闭合。当打开床头皮带罩后，SQ1 的常开触头断开，切断控制电路电源，以确保人身安全。钥匙开关 SB 和行程开关 SQ2 在车床正常工作时是断开的，QF 的线圈不通电，断路器 QF 能合闸。当打开配电壁龛门时，SQ2 闭合，QF 线圈获电，断路器 QF 自动断开，切断车床的电源。

1）主轴电动机 M1 的控制。



2）冷却泵电动机 M2 的控制。主轴电动机 M1 和冷却泵电动机 M2 在控制电路中实现顺序控制，只有当主轴电动机 M1 起动后，KM 的常开触头闭合，合上旋钮开关 SB4，中间继电器 KA1 吸合，冷却泵电动机 M2 才能起动。当 M1 停止运行或断开旋钮开关 SB4 时，M2 停止运行。

3）刀架快速移动电动机 M3 的控制。刀架快速移动电动机 M3 的起动是由安装在进给操作手柄顶端的按钮 SB3 控制的，它与中间继电器 KA2 组成点动控制环节。将操作手柄扳到所需移动的方向，按下 SB3，KA2 得电吸合，电动机 M3 起动运转，刀架沿指定的方向快速移动。刀架快速移动电动机 M3 是短时间工作的，故未设过载保护。

（3）照明与信号电路分析

控制变压器 TC 的二次侧输出 24V 和 6V 电压，分别作为车床低压照明和指示灯的电源。EL 为车床的低压照明灯，由开关 SA 控制，FU4 做短路保护；HL 为电源指示灯，FU3 做短路保护。

3. 用 PLC 改造 CA6140 车床

（1）PLC 控制系统的主电路接线图

图 7.1-3 为 CA6140 车床 PLC 改造的主电路图。

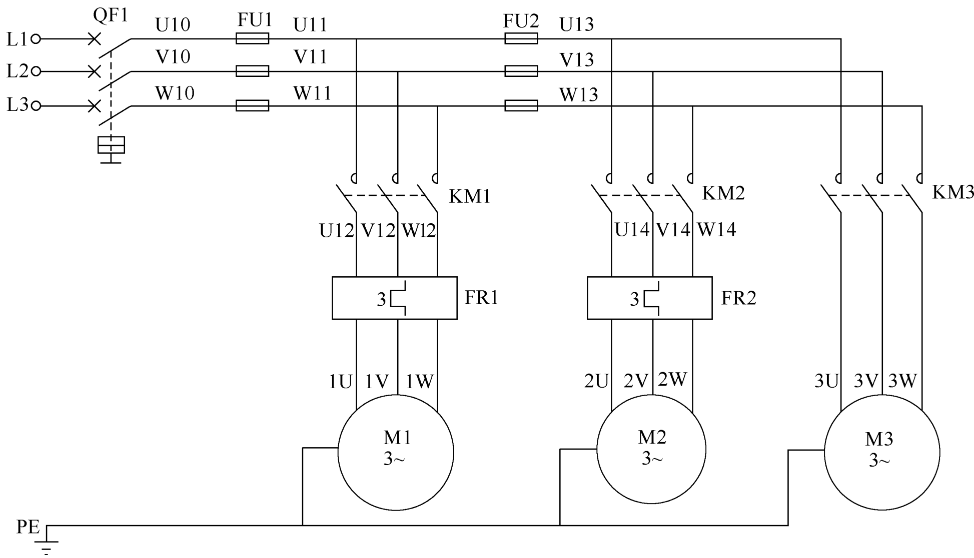


图 7.1-3　CA6140 车床 PLC 改造的主电路图

（2）PLC 控制系统的 I/O 接线

1）分配 PLC 的 I/O 地址通道。根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 10 个输入点，6 个输出点，见表 7.1-2。

表 7.1-2　PLC 的 I/O 配置

输入设备			输出设备		
代号	功能	输入继电器	代号	功能	输出继电器
SB	钥匙开关	I0.0	QF	QF 的线圈	Q0.0
SB1	停止 M1	I0.1	KM1	控制 M1	Q0.1

（续）

输入设备			输出设备		
代号	功能	输入继电器	代号	功能	输出继电器
SB2	起动 M1	I0.2	KM2	控制 M2	Q0.2
SB3	起动 M3	I0.3	KM3	控制 M3	Q0.3
SA1	控制 M2	I0.4	HL	刻度照明	Q0.4
SA2	照明灯开关	I0.5	EL	工作照明	Q0.5
SQ1	传动带罩防护开关	I0.6			
SQ2	电气箱防护开关	I0.7			
FR1	M1 过载保护	I1.0			
FR2	M2 过载保护	I1.1			

2）PLC 控制系统的 I/O 接线图。根据控制要求，设计并绘制 PLC 控制系统的 I/O 接线原理图，如图 7.1-4 所示。

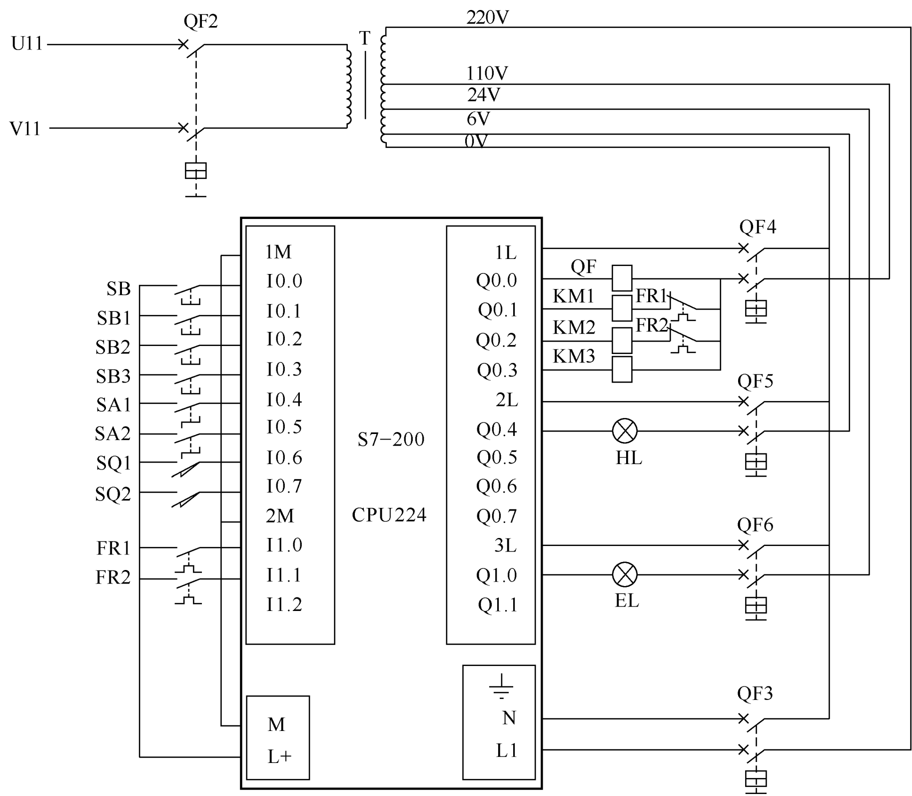


图 7.1-4　PLC 控制系统 I/O 接线图

（3）安装与接线

1）将所有元件装在一块配电板上，做到布局合理、安装牢固、符合安装工艺规范。

2）根据接线原理图配线，做到接线正确、牢固、美观。

3）I/O 线和动力线应分开走线，并保持距离。数字量信号一般采用普通电缆就可以；模拟信号线和高速信号线应采用屏蔽电缆，并做好接地工作。

4）安装 PLC 应远离强干扰源，并可靠接地，最好和强电的接地装置分开，接地线的截面积应大于 $2mm^2$，接地点与 PLC 的距离应小于 50cm。

（4）程序设计

CA6140 车床的 PLC 梯形图程序如图 7.1-5 所示。

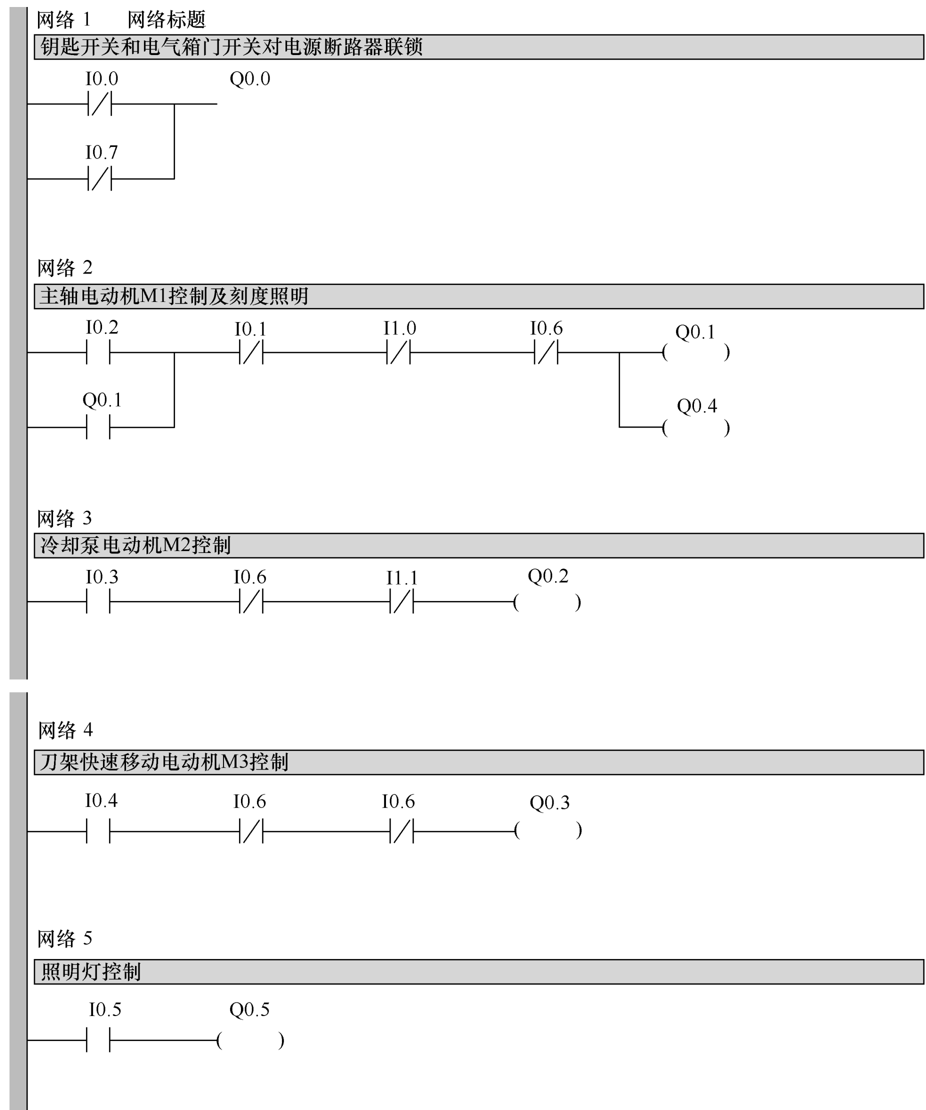


图 7.1-5　PLC 梯形图程序

（5）程序输入与调试

熟练地操作编程软件，能正确将编制的程序输入 PLC；按照被控设备的要求进行调试、修改，达到设计要求。

1）通电前使用万用表检查电路的正确性，确保通电成功。

2）调试程序时先对程序进行模拟调试，对系统各种工作要求和方式都要逐一检查，不能遗漏，直到符合控制要求。

3）现场调试中，接入实际的信号和负载时，应充分考虑各种可能的情况，做到认真、仔细、全面地完成现场调试。

4）注意人身和设备的安全。

实例 2　X62W 万能铣床的 PLC 控制

1. X62W 万能铣床的控制要求

万能铣床是一种通用的多用途机床，它可以用圆柱铣刀、圆片铣刀、角度铣刀、成型铣刀及端面铣刀等刀具对各种零件进行平面、斜面、螺旋面及成型表面的加工，还可以加装万能铣头、分度头和圆工作台等机床附件来扩大加工范围。

常用的万能铣床有两种：一种是 X62W 型卧式万能铣床，铣头水平方向放置；另一种是 X52K 型立式万能铣床，铣头垂直方向放置。

X62W 万能铣床主要由底座、床身、悬梁、主轴、刀杆支架、工作台、回转盘、横溜板和升降台等组成，如图 7.2-1 所示。其主要运动形式及控制要求如下。

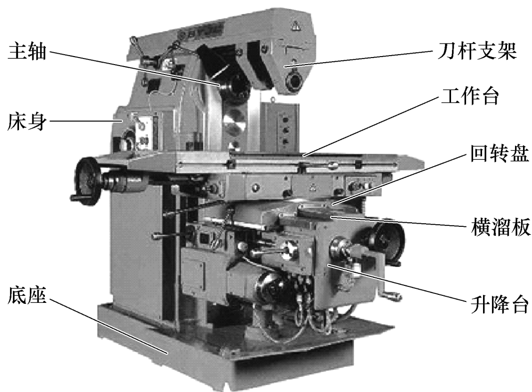


图 7.2-1　X62W 万能铣床外形结构

（1）主运动

X62W 万能铣床的主运动是主轴带动铣刀的旋转运动。

铣削加工有顺铣和逆铣两种加工方式，所以要求主轴电动机能正反转。但考虑到大多数情况下一批或多批工件只用一个方向铣削，在加工过程中不需要变换主轴旋转的方向，因此用组合开关来控制主轴电动机的正反转。

铣削加工是一种不连续的切削加工方式，为减小振动，主轴上装有惯性轮，但这样会造成主轴停车困难，为此主轴电动机采用电磁离合器制动以实现准确停车。

铣削加工过程中需要主轴调速，采用改变变速箱的齿轮传动比来实现，主轴电动机不需要调速。

（2）进给运动

进给运动是指工件随工作台在前后、左右和上下六个方向上的运动以及随圆形工作台的旋转运动。

铣床的工作台要求有前后、左右和上下六个方向上的进给运动和快速移动，所以要求进给电动机能正反转。为扩大加工能力，在工作台上可加装圆形工作台，圆形工作台的回转运动是由进给电动机经传动机构驱动的。

为保证机床和刀具的安全，在铣削加工时，任何时刻工件都只能有一个方向的进给运动，因此采用了机械操作手柄和行程开关相配合的方式实现六个运动方向的联锁。

为防止刀具和机床的损坏，要求只有主轴旋转后才允许有进给运动和进给方向的快速移动；同时为了减小加工件的表面粗糙度，要求进给停止后主轴才能停止或同时停止。

进给变速采用机械方式实现，进给电动机不需要调速。

（3）辅助运动

辅助运动包括工作台的快速运动及主轴和进给的变速冲动。

工作台的快速运动是指工作台在前后、左右和上下六个方向之一上的快速移动。它是通过快速移动电磁离合器的吸合，改变机械传动链的传动比实现的。

为保证变速后齿轮能良好啮合，主轴和进给变速后，都要求电动机做瞬时点动，即变速冲动。

2. X62W 万能铣床电气控制电路分析

X62W 万能铣床的电路图如图 7.2-2 所示，它分为主电路、控制电路和照明电路三部分。

（1）主电路分析

主电路共有三台电动机，其控制和保护电器见表 7.2-1。

表 7.2-1　主电路的控制与保护电器

名称及代号	功　　能	控 制 电 器	过载保护电器	短路保护电器
主轴电动机 M1	拖动主轴带动铣刀旋转	接触器 KM1 和组合开关 SA	热继电器 FR1	熔断器 FU1
进给电动机 M2	拖动进给运动和快速移动	接触器 KM3 和 KM4	热继电器 FR3	熔断器 FU1
冷却泵电动机 M3	供应冷却液	手动开关 QS2	热继电器 FR2	熔断器 FU1

（2）控制电路分析

控制电路的电源由控制变压器 TC 输出 110V 电压供电。

1）主轴电动机 M1 的控制。为方便操作，主轴电动机 M1 采用两地控制方式，一组起动按钮 SB1 和停止按钮 SB5 安装在工作台上，另一组起动按钮 SB2 和停止按钮 SB6 安装在床身上。主轴电动机 M1 的控制包括起动控制、制动控制、换刀控制和变速冲动控制，具体见表 7.2-2。

表 7.2-2　主轴电动机 M1 的控制

控 制 要 求	控 制 作 用	控 制 过 程
起动控制	起动主轴电动机 M1	选择好主轴的转速和转向，按下起动按钮 SB1 或 SB2，接触器 KM1 得电吸合并自锁，M1 起动运转，同时 KM1 的辅助常开触头（9－10）闭合，为工作台进给电路提供电源
制动控制	停车时使主轴迅速停转	按下停止按钮 SB5（或 SB6），其常闭触头 SB5-1 或 SB6-1（13 区）断开，接触器 KM1 线圈断电，KM1 的主触头分断，电动机 M1 断电做惯性运转；常开触头 SB5-2 或 SB6-2（8 区）闭合，电磁离合器 YC1 通电，M1 制动停转
换刀控制	更换铣刀时将主轴制动，以方便换刀	将转换开关 SA1 扳向换刀位置，其常开触头 SA1-1（8 区）闭合，电磁离合器 YC1 得电将主轴制动；同时常闭触头 SA1-2（13 区）断开，切断控制电路，铣床不能通电运转，确保人身安全
变速冲动控制	保证变速后齿轮能良好啮合	变速时先将变速手柄向下压并向外拉出，转动变速盘选定所需转速后，将手柄推回。此时冲动开关 SQ1（13 区）短时受压，主轴电动机 M1 点动，手柄推回原位后，SQ1 复位，M1 断电，变速冲动结束

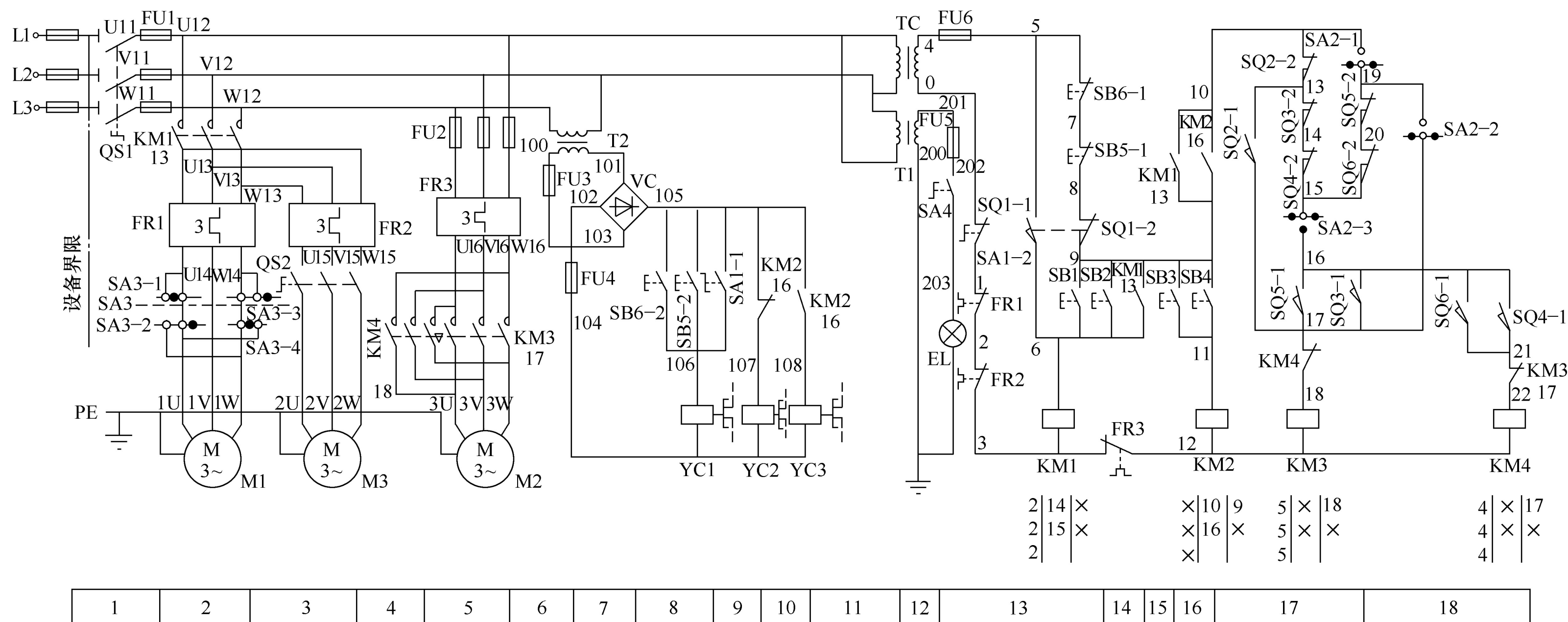


图 7.2-2　X62W 万能铣床的电路图

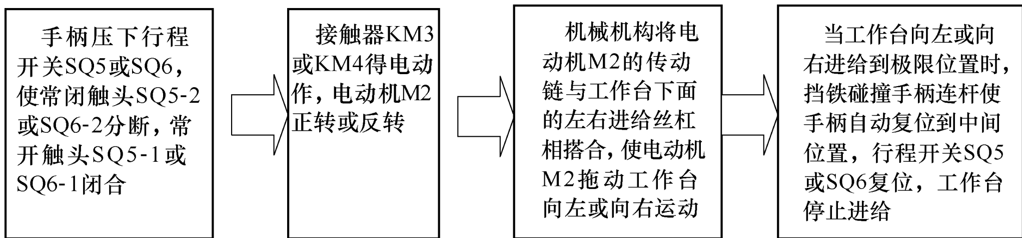
2）进给电动机 M2 的控制。铣床的工作台要求有前后、左右和上下六个方向上的进给运动和快速移动，并且可在工作台上安装附件圆形工作台，进行对圆弧或凸轮的铣削加工。这些运动都是由进给电动机 M2 拖动的。

① 工作台前后、左右和上下六个方向上的进给运动。工作台的前后和上下进给运动由一个手柄控制，左右进给运动由另一个手柄控制。手柄位置与工作台运动方向的关系见表 7.2-3。

表 7.2-3　控制手柄的位置与工作台运动方向的关系

控 制 手 柄	手 柄 位 置	行程开关动作	接触器动作	电动机 M2 转向	传动链搭合丝杠	工作台运动方向
左右进给手柄	左	SQ5	KM3	正转	左右进给丝杠	向左
	中			停止		停止
	右	SQ6	KM4	反转	左右进给丝杠	向右
上下和前后进给手柄	上	SQ4	KM4	反转	上下进给丝杠	向上
	下	SQ3	KM3	正转	上下进给丝杠	向下
	中			停止		
	前	SQ3	KM3	正转	前后进给丝杠	向前
	后	SQ4	KM4	反转	前后进给丝杠	向后

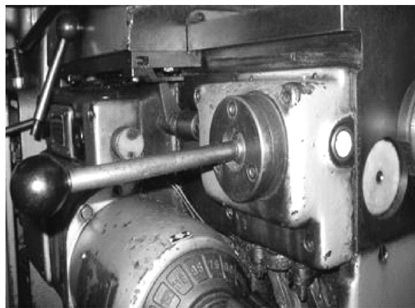
下面以工作台的左右移动为例分析工作台的进给。左右进给操作手柄与行程开关 SQ5 和 SQ6 联动，有左、中、右三个位置，其控制关系见表 7.2-3。当手柄扳向中间位置时，行程开关 SQ5 和 SQ6 均未被压合，进给控制电路处于断开状态；当手柄扳向左（或右）位置时，如图 7.2-3 所示，手柄压下行程开关 SQ5（或 SQ6），同时将电动机的传动链和左右移动丝杠相连。控制过程如下：



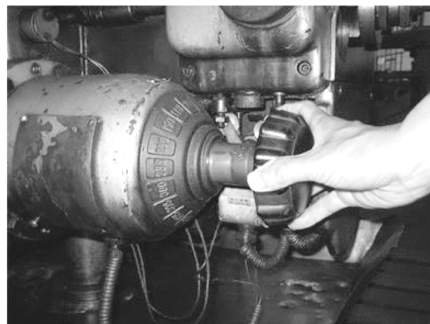
工作台的上下和前后进给由上下和前后进给手柄控制，如图 7.2-4 所示，其控制过程与左右进给相似，这里不再一一分析。通过以上分析可见，两个操作手柄被置定于某一方向后，只能压下四个行程开关 SQ3、SQ4、SQ5、SQ6 中的一个，接通电动机 M2 正转或反转电路，同时通过机械机构将电动机的传动链与三根丝杠（左右丝杠、上下丝杠、前后丝杠）中的一根（只能是一根）相搭合，拖动工作台沿选定的进给方向运动，而不会沿其他方向运动。

② 左右进给与上下前后进给的联锁控制。在控制进给的两个手柄中，当其中的一个操作手柄被置定在某一进给方向后，另一个操作手柄必须置于中间位置，否则将无法实现任何进给运动。这是因为在控制电路中对两者实行了联锁保护。如当把左右进给手柄扳向左时，若又将另一个进给手柄扳到向下进给方向，则行程开关 SQ5 和 SQ3 均被压下，触头 SQ5-2 和 SQ3-2 均分断，断开了接触器 KM3 和 KM4 的通路，电动机 M2 只能停转，保证了操作安全。

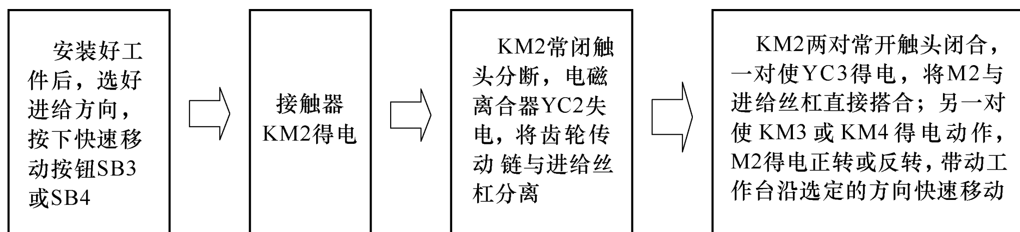

图 7.2-3　左右进给图


图 7.2-4　上下与前后进给手柄

③ 进给变速时的瞬时点动。和主轴变速时一样，进给变速时，为使齿轮进入良好的啮合状态，也要进行变速后的瞬时点动。进给变速时，必须先把进给操纵手柄放在中间位置，然后将进给变速盘（在升降台前面）向外拉出，选择好速度后，再将变速盘推进去，如图 7.2-5 所示。在推进的过程中，挡块压下行程开关 SQ2，使触头 SQ2-2 分断，SQ2-1 闭合，接触器 KM3 经 10—19—20—15—14—13—17—18 路径得电动作，电动机 M2 起动；但随着变速盘复位，行程开关 SQ2 跟着复位，使 KM3 断电释放，M2 失电停转。这样使电动机 M2 瞬时点动一下，齿轮系统产生一次抖动，齿轮便顺利啮合了。

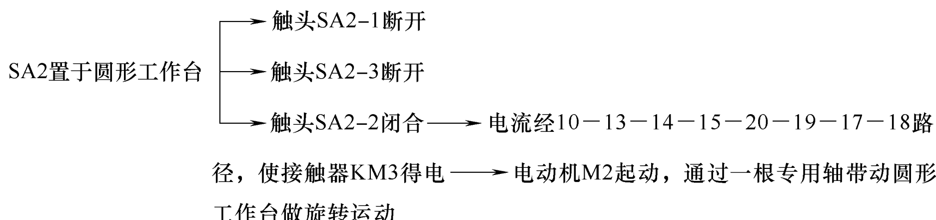

图 7.2-5　进给变速冲动

④ 工作台的快速移动控制。快速移动是通过两个进给操作手柄和快速移动按钮 SB3 或 SB4 配合实现的。控制过程如下：



松开 SB3 或 SB4，快速移动停止。

⑤ 圆形工作台的控制。圆形工作台的工作由转换开关 SA2 控制。当需要圆形工作台旋转时，将开关 SA2 扳到接通位置，此时：



当不需要圆形工作台旋转时，转换开关 SA2 扳到断开位置，这时触头 SA2-1 和 SA2-3 闭合，触头 SA2-2 断开，工作台在六个方向上正常进给，圆形工作台不能工作。

圆形工作台开动时其余进给一律不准运动，两个进给手柄必须置于零位。若出现误操作，扳动两个进给手柄中的任意一个，则必然压合行程开关 SQ3～SQ6 中的一个，使电动机停止转动。圆形工作台加工不需要调速，也不要求正反转。

3）冷却泵及照明电路的控制。主轴电动机 M1 和冷却泵电动机 M3 采用的是顺序控制，即只有在主轴电动机 M1 起动后冷却泵电动机 M3 才能起动。冷却泵电动机 M3 由组合开关 QS2 控制。

机床照明由变压器 T1 供给 24V 的安全电压，由开关 SA4 控制。熔断器 FU5 做照明电路的短路保护。

3. 用 PLC 改造 X62W 万能铣床

（1）PLC 控制系统的主电路接线图

图 7.2-6 为 X62W 万能铣床 PLC 改造的主电路图。

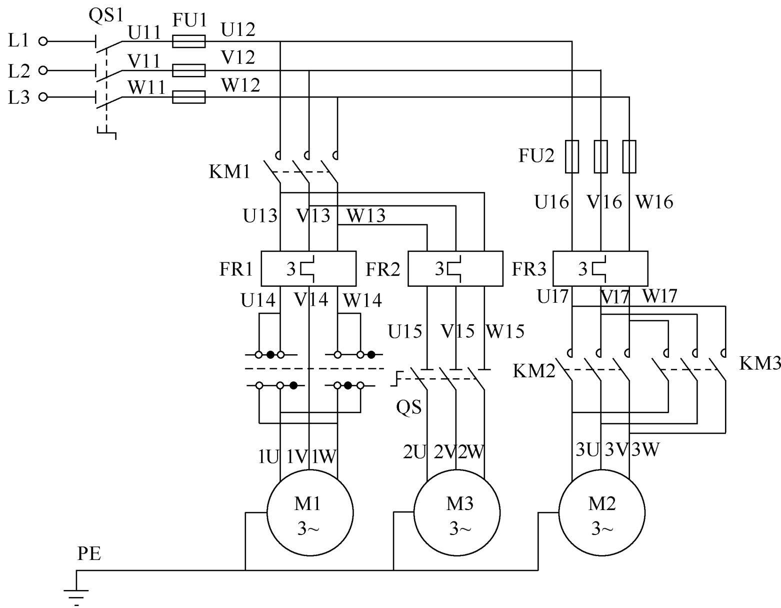


图 7.2-6　X62W 万能铣床 PLC 改造的主电路图

（2）PLC 控制系统的 I/O 接线

1）分配 PLC 的 I/O 地址通道。根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 15 个输入点，7 个输出点，见表 7.2-4。

表 7.2-4　PLC 的 I/O 配置

输入设备			输出设备		
代号	功能	输入继电器	代号	功能	输出继电器
SB1、SB2	主轴电动机 M1 起动	I0.0	KM1	控制主轴 M1 起停	Q0.0
SB3、SB4	快速进给点动	I0.1	KM2	控制进给 M2 正转	Q0.1
SB5、SB6	主轴电动机 M1 停止、制动	I0.2	KM3	控制进给 M2 反转	Q0.2

（续）

输入设备			输出设备		
代号	功能	输入继电器	代号	功能	输出继电器
SA1	换刀开关	I0.3	YC1	主轴 M1 制动控制	Q0.4
SA2	圆形工作台开关	I0.4	YC2	M2 正常进给	Q0.5
SQ1	主轴冲动开关	I0.5	YC3	M2 快速进给	Q0.6
SQ2	进给冲动开关	I0.6	EL	工作照明灯	Q1.0
SQ3	M2 正反转及联锁	I0.7			
SQ4	M2 正反转及联锁	I1.0			
SQ5	M2 正反转及联锁	I1.1			
SQ6	M2 正反转及联锁	I1.2			
FR1	M1 过载保护	I1.3			
FR2	M3 过载保护	I1.4			
FR3	M2 过载保护	I1.5			
SA4	工作照明灯开关	I1.6			

2）PLC 控制系统的 I/O 接线图。根据控制要求，设计并绘制 PLC 控制系统的 I/O 接线原理图，如图 7.2-7 所示。

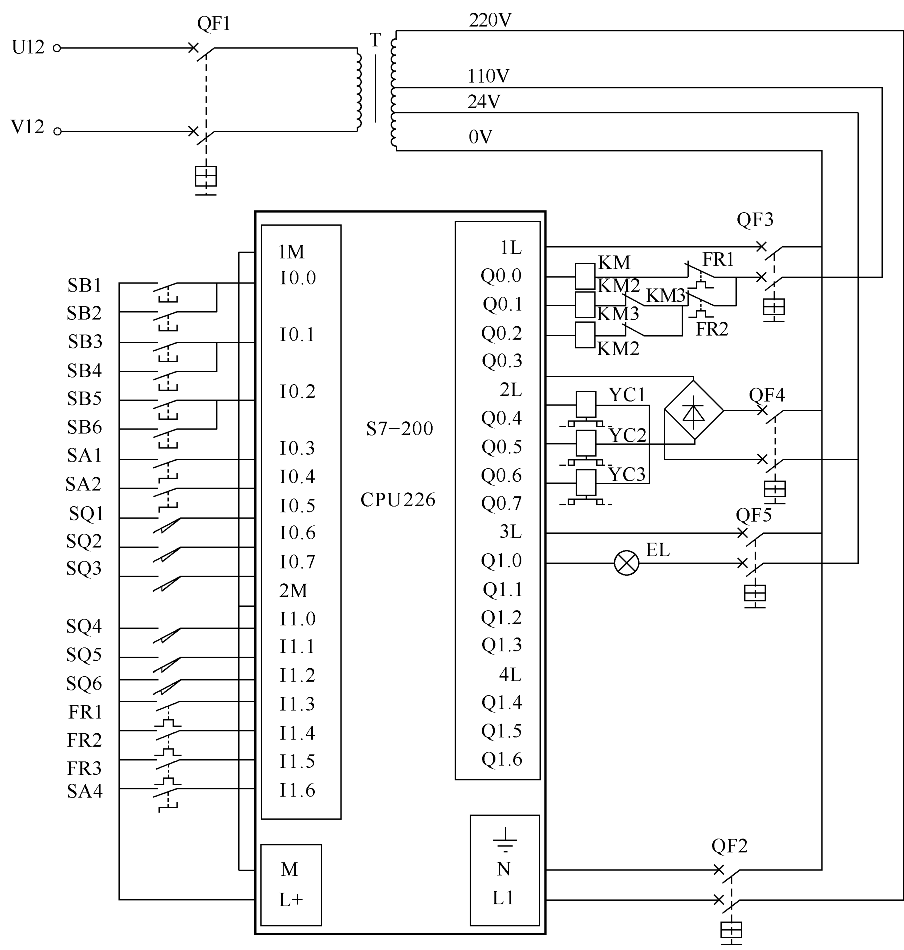


图 7.2-7　PLC 控制系统 I/O 接线图

（3）安装与接线

1）将所有元件装在一块配电板上，做到布局合理、安装牢固、符合安装工艺规范。

2）根据接线原理图配线，做到接线正确、牢固、美观。

3）I/O 线和动力线应分开走线，并保持距离。数字量信号一般采用普通电缆就可以；模拟信号线和高速信号线应采用屏蔽电缆，并做好接地工作。

4）安装 PLC 应远离强干扰源，并可靠接地，最好和强电的接地装置分开，接地线的截面积应大于 $2mm^2$，接地点与 PLC 的距离应小于 50cm。

（4）程序设计

X62W 铣床的 PLC 梯形图程序如图 7.2-8 所示。

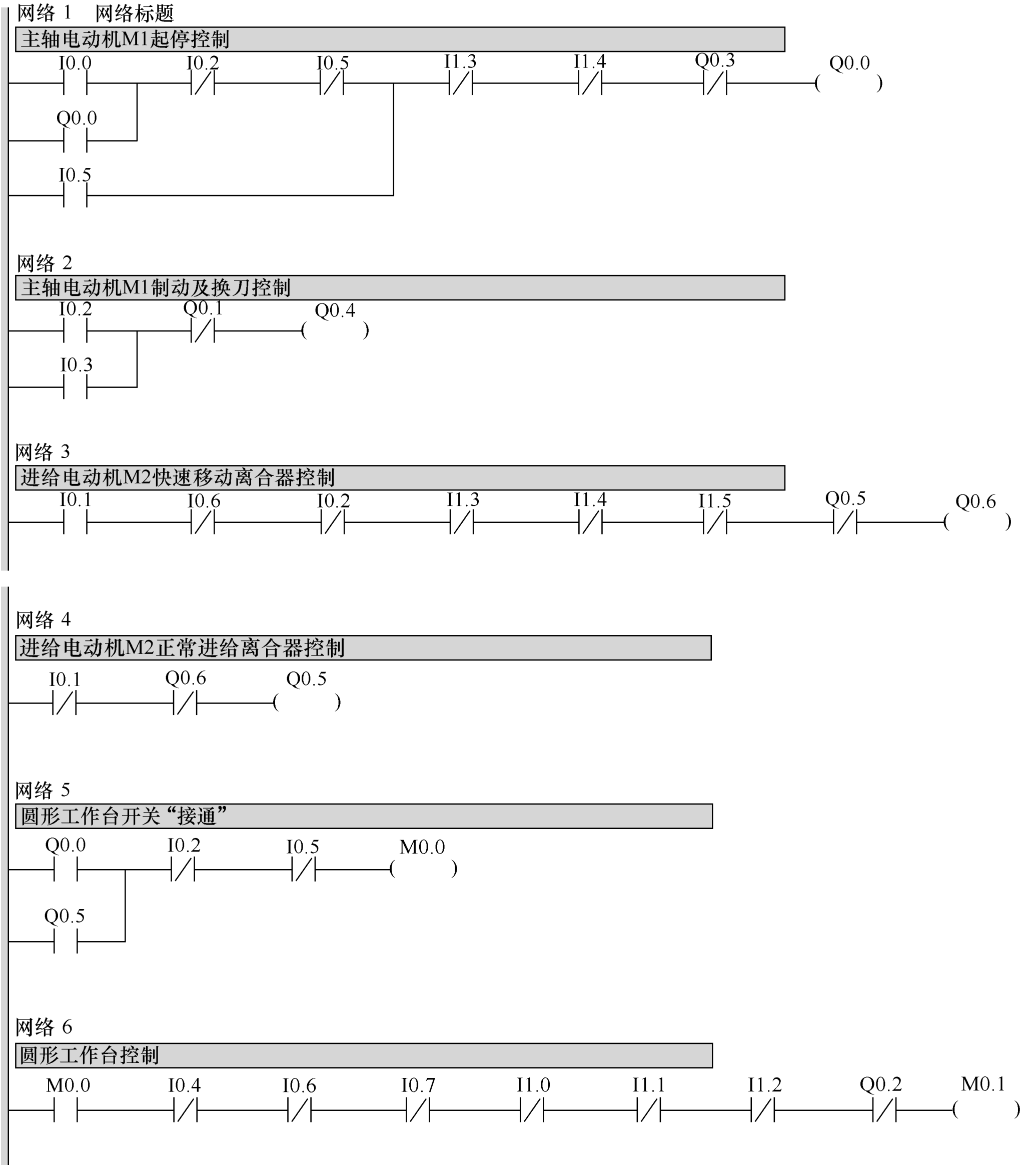


图 7.2-8 X62W 铣床的 PLC 梯形图程序

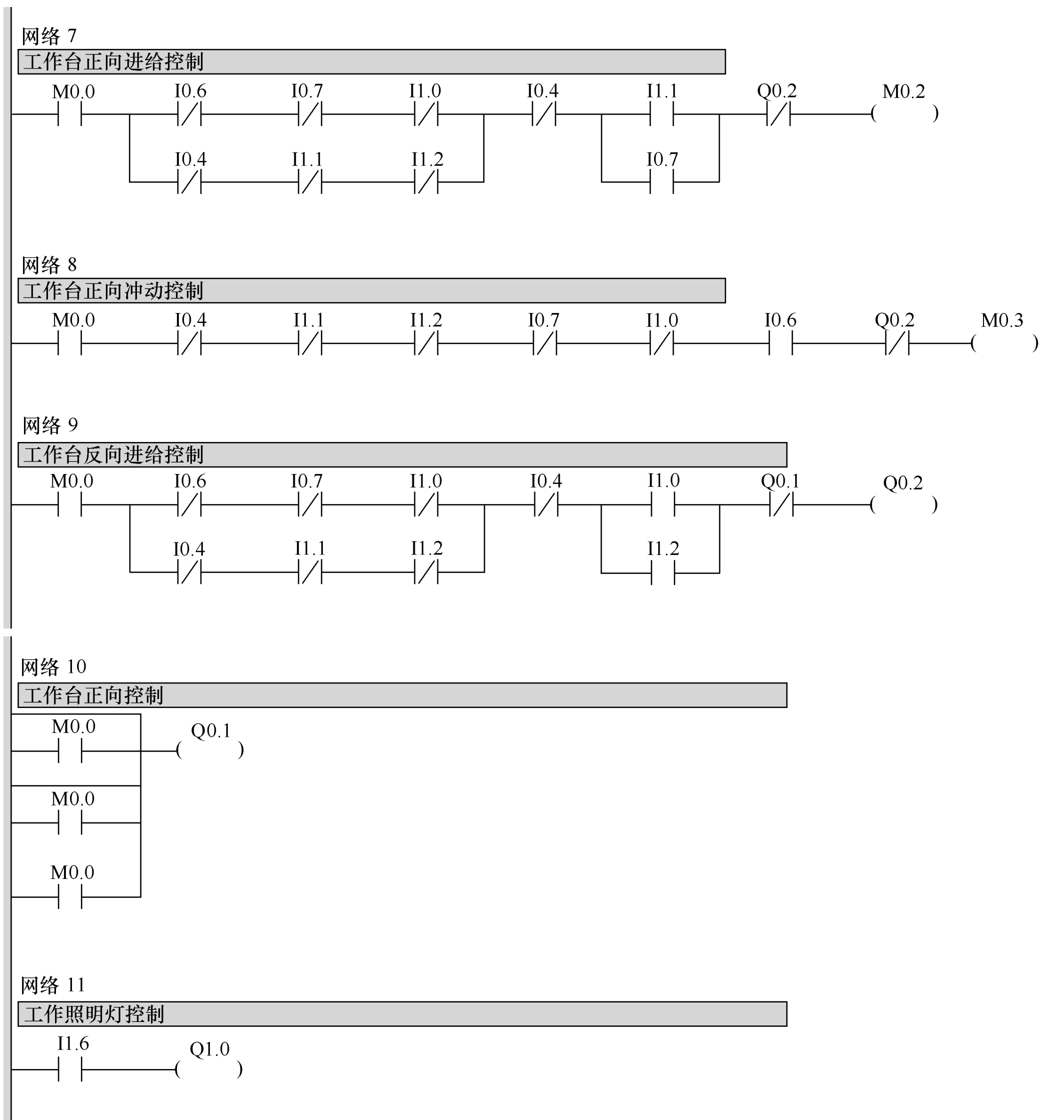


图 7. 2-8 X62W 铣床的 PLC 梯形图程序（续）

（5）程序输入与调试

熟练地操作编程软件，能正确将编制的程序输入 PLC；按照被控设备的要求进行调试、修改，达到设计要求。

1）通电前使用万用表检查电路的正确性，确保通电成功。

2）调试程序时先对程序进行模拟调试，对系统各种工作要求和方式都要逐一检查，不能遗漏，直到符合控制要求。

3）现场调试中，接入实际的信号和负载时，应充分考虑各种可能的情况，做到认真、仔细、全面地完成现场调试。

4）注意人身和设备的安全。

实例 3　Z3040 摇臂钻床的 PLC 控制

摇臂钻床利用旋转的钻头对工件进行加工，其由底座、立柱、摇臂、主轴箱和工作台等组成，如图 7.3-1 所示。主轴箱固定在摇臂上，可以沿摇臂径向运动。摇臂借助于丝杠做升降运动，也可以与外立柱固定在一起沿内立柱旋转。钻削加工时，通过夹紧机构将主轴箱紧固在摇臂上，摇臂紧固在立柱上。

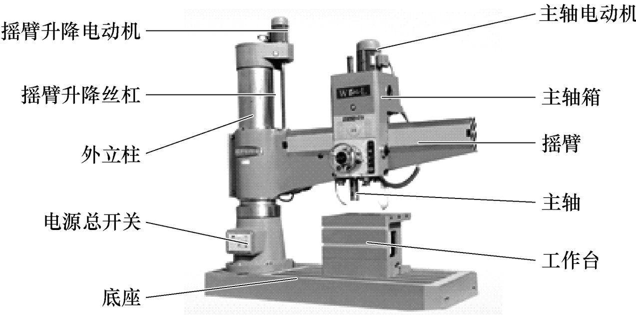


图 7.3-1　Z3040 摇臂钻床外形

1. Z3040 摇臂钻床的控制要求

1）Z3040 摇臂钻床相对运动部件较多，为简化传动装置，采用多台电动机拖动。

2）各种工作状态都通过十字开关 SA 操作，为防止十字开关手柄停在任何工作位置时，因接通电源而产生误动作，本控制电路设有零电压保护环节。

3）摇臂升降要求有限位保护。

4）钻削加工时需要对刀具及工件进行冷却。各电动机功能及控制要求见表 7.3-1。

表 7.3-1　各电动机功能及控制要求

电动机名称及代号	作　用	控制要求
主轴电动机 M1	拖动钻削及进给运动	单向运转，主轴的正反转通过摩擦离合器实现
摇臂升降电动机 M2	拖动摇臂升降	正反转控制，通过机械和电气联合控制
液压泵电动机 M3	拖动内、外立柱及主轴箱与摇臂夹紧与放松	正反转控制，通过液压装置和电气联合控制
冷却泵电动机 M4	供给冷却液	正转控制，拖动冷却泵输送冷却液

2. Z3040 摇臂钻床电气控制电路分析

图 7.3-2 为 Z3040 钻床的电气控制电路图。

（1）主电路介绍

该钻床共有四台电动机。

主轴电动机 M1：提供主轴旋转的动力，由交流接触器 KM1 控制单向运转，热继电器 FR1 做过载保护，断路器 QF1 兼做短路保护。

摇臂升降电动机 M2：提供摇臂升降的动力，由交流接触器 KM2 和 KM3 控制 M2 正反转，用于间歇工作未设过载保护，断路器 QF3 做短路保护。

液压泵电动机 M3：提供液压油，用于摇臂、立柱、主轴箱的夹紧和松开，由交流接触器 KM4 和 KM5 控制 M3 正反转，热继电器 FR2 做过载保护，断路器 QF3 做短路保护。

冷却泵电动机 M4：输送冷却液，由断路器 QF2 控制并兼做过载保护。

Z3040 钻床辅助控制有主轴运转指示灯、照明灯、电源指示灯。

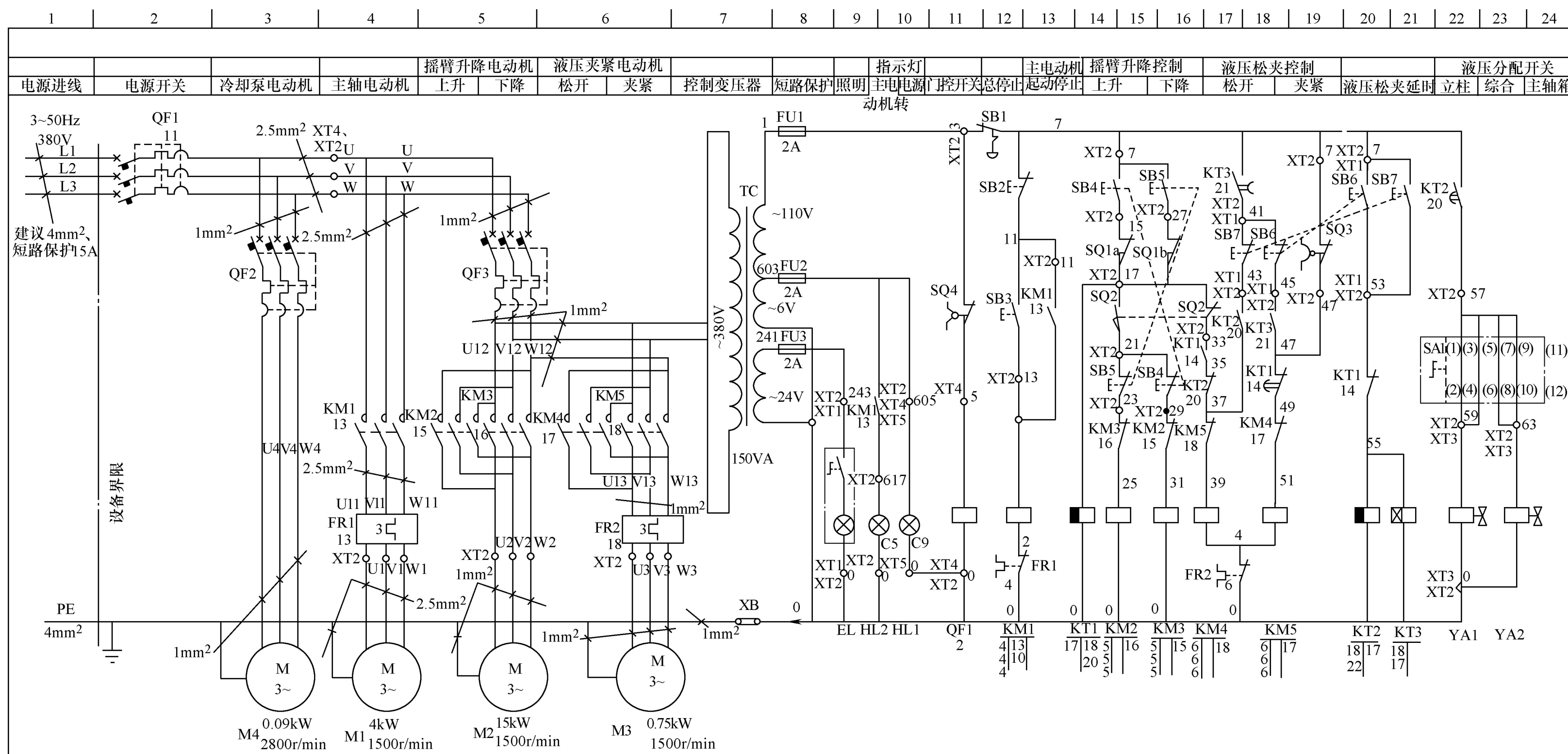


图 7.3-2　Z3040 摇臂钻床的电气控制线路图

（2）控制电路介绍

控制电路电源由控制变压器 TC 输出 110V 电压提供，熔断器 FU1 作为短路保护。电源指示灯为 6.3V 电压，局部照明灯为 24V，变压器二次侧设有熔断器做短路保护，SB1 为急停按钮，按下 SB1 整机停止工作，防止意外事故，SQ4 位置开关是控制柜门开门断电所设定的。

（3）控制电路原理分析

1）主轴电动机 M1 的控制。按下起动按钮 SB3，接触器 KM1 吸合并自锁，M1 起动运转，指示灯 HL2 亮；按下停止按钮 SB2，接触器 KM1 释放，M1 电动机停止运转，指示灯 HL2 熄灭。

2）摇臂升降控制。

① 摇臂上升控制：按下上升按钮 SB4，断电延时继电器 KT1 得电吸合，其常开瞬时触点闭合，常闭瞬时触点断开，接触器 KM4 得电吸合，液压泵电动机 M3 起动正转，通过液压传动机构，使 SQ2 位置开关常闭触头断开，常开触头闭合，前者切断 KM4 电路，液压泵电动机 M3 停转，后者使 KM2 线圈吸合，KM2 主触头闭合，接通电动机 M2 运转摇臂上升；上升到需要位置，松开按钮 SB4，KM2 和时间继电器 KT1 同时断电释放，电动机 M2 停转，上升停止，KT1 恢复初始状态。

② 摇臂下降控制：按下下降按钮 SB5，断电延时继电器 KT1 得电吸合，其瞬时触点断开，这时液压泵电动机运转原理与上升一样，当到达需要位置时松开 SB5，KM3 线圈失电，电动机 M2 停转，由于时间断电器 KT1 断电释放，经 2～3s 时间延时，延时触点闭合，电源 110V 电压使 KM5 线圈得电吸合，电动机 M3 反转，通过液压装置使摇臂夹紧，夹紧后通过机械传动，碰触到位置开关使常闭触头断开，KM5 断电释放，电动机 M3 停转，完成了松开—上升或下降—夹紧的过程。

位置开关 SQ1a 为上升限位保护，SQ1b 为下降限位保护；位置开关 SQ3 是自动夹紧的关键电器，调整不当或常闭触头断不开都易使电动机 M3 过载。摇臂升降电动机 M2 的正反转采用电气双重联锁保护，为防止主电路故障，确保电路安全工作。

3）立柱和主轴箱的夹紧与放松控制。立柱和主轴箱的夹紧（或放松）既可以同时进行，也可以单独进行，由转换开关 SA1 和复合按钮 SB6、SB7 控制。SA1 有三个位置，中间位置为立柱和主轴箱的夹紧与放松同时进行，左边位置为立柱和主轴箱的单独夹紧与放松，右边位置为主轴箱的夹紧或放松。复合按钮 SB6 是松开按钮，SB7 是夹紧按钮。

① 立柱和主轴箱同时松开与夹紧。将转换开关 SA1 拨到中间位置，按下按钮 SB6，时间继电器 KT2、KT3 线圈得电吸合，KT2 的延时断开常开触点瞬时闭合，电磁铁 YA1、YA2 得电吸合，KT3 的延时闭合常开触点经 1～3s 延时后闭合，接触器 KM4 得电吸合，液压泵电动机 M3 正转。

松开按钮 SB6，时间继电器 KT2 和 KT3 线圈断电释放，KT3 延时闭合的常开触点瞬时分断，KM4 断电释放，KT2 延时分断的常开触点经 1～3s 后分断，电磁铁 YA1、YA2 断电释放，立柱和主轴箱同时松开。

立柱和主轴箱同时夹紧的工作原理与松开相似，按下按钮 SB7，接触器 KM5 得电吸合，电动机 M3 反转，可自己分析。

② 立柱和主轴箱单独松开与夹紧。如果需要单独控制立柱，可将转换开关 SA1 扳至左

侧位置。按下夹紧按钮 SB7，时间继电器 KT2 和 KT3 线圈同时得电吸合，电磁铁 YA1 得电吸合，立柱夹紧。

松开按钮 SB7，时间继电器 KT2 和 KT3 线圈断电释放，KT3 的通电延时闭合常开触点瞬时断开，接触器 KM5 线圈断电释放，液压泵电动机 M3 停转，经 1～3s 的延时后，KT2 延时分断的常开触点分断，电磁铁 YA1 失电释放，立柱夹紧操作结束。

同理，主轴箱的松开可自行分析。

3. 用 PLC 改造 Z3040 摇臂钻床

（1）PLC 控制系统的主电路接线图

图 7.3-3 为 Z3040 钻床 PLC 改造的主电路图。

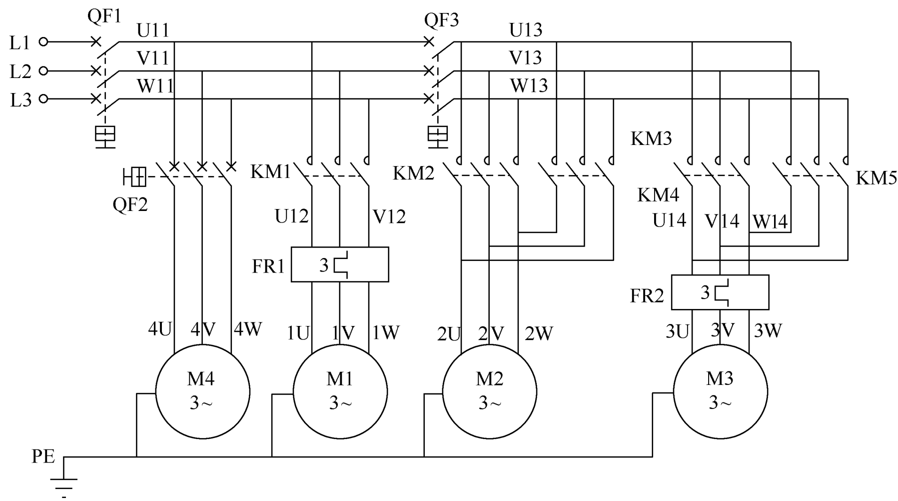


图 7.3-3　Z3040 钻床 PLC 控制系统的主电路图

（2）PLC 控制系统的 I/O 接线

1）分配 PLC 的 I/O 地址通道。根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 12 个输入点，6 个输出点，见表 7.3-2。

表 7.3-2　PLC 的 I/O 配置

输入设备			输出设备		
代号	功能	输入继电器	代号	功能	输出继电器
SB1	主轴电动机 M1 停止	I0.0	KM1	控制主轴电动机 M1	Q0.0
SB2	主轴电动机 M1 起动	I0.1	KM2	控制摇臂电动机 M2 正转	Q0.1
SB3	摇臂上升按钮	I0.2	KM3	控制摇臂电动机 M2 反转	Q0.2
SB4	摇臂下降按钮	I0.3	KM4	控制液压电动机 M3 正转	Q0.3
SB5	松开控制按钮	I0.4	KM5	控制液压电动机 M3 反转	Q0.4
SB6	夹紧控制按钮	I0.5	YA	液压控制电磁阀	Q0.5
SQ1	摇臂上升限位（接常闭触点）	I0.6			
SQ2	摇臂下降限位（接常闭触点）	I0.7			
SQ3	摇臂松开限位	I1.0			
SQ4	摇臂夹紧限位	I1.1			
FR1	M1 过载保护	I1.2			
FR2	M3 过载保护	I1.3			

2）PLC 控制系统的 I/O 接线图。根据控制要求，设计并绘制 PLC 控制系统的 I/O 接线原理图，如图 7.3-4 所示。

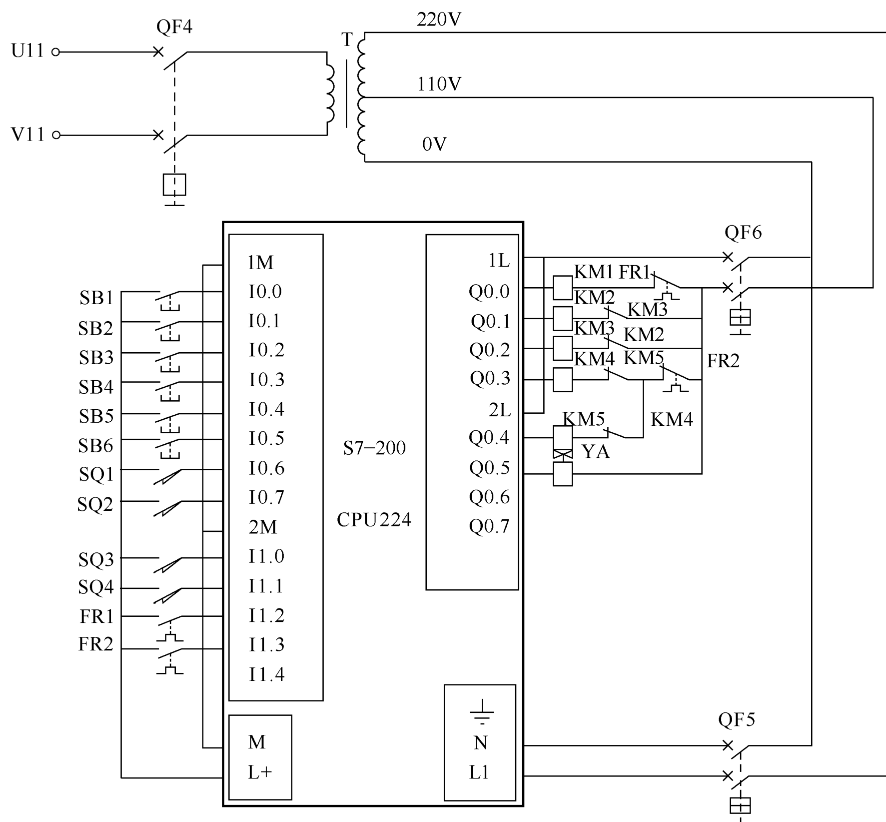


图 7.3-4　PLC 控制系统 I/O 接线图

（3）安装与接线

1）将所有元件装在一块配电板上，做到布局合理、安装牢固、符合安装工艺规范。

2）根据接线原理图配线，做到接线正确、牢固、美观。

3）I/O 线和动力线应分开走线，并保持距离。数字量信号一般采用普通电缆就可以；模拟信号线和高速信号线应采用屏蔽电缆，并做好接地工作。

4）安装 PLC 应远离强干扰源，并可靠接地，最好和强电的接地装置分开，接地线的截面积应大于 $2mm^2$，接地点与 PLC 的距离应小于 50cm。

（4）程序设计

Z3040 钻床的 PLC 梯形图程序如图 7.3-5 所示。

（5）程序输入与调试

熟练地操作编程软件，能正确将编制的程序输入 PLC；按照被控设备的要求进行调试、修改，达到设计要求。

1）通电前使用万用表检查电路的正确性，确保通电成功。

2）调试程序时先对程序进行模拟调试，对系统各种工作要求和方式都要逐一检查，不能遗漏，直到符合控制要求。

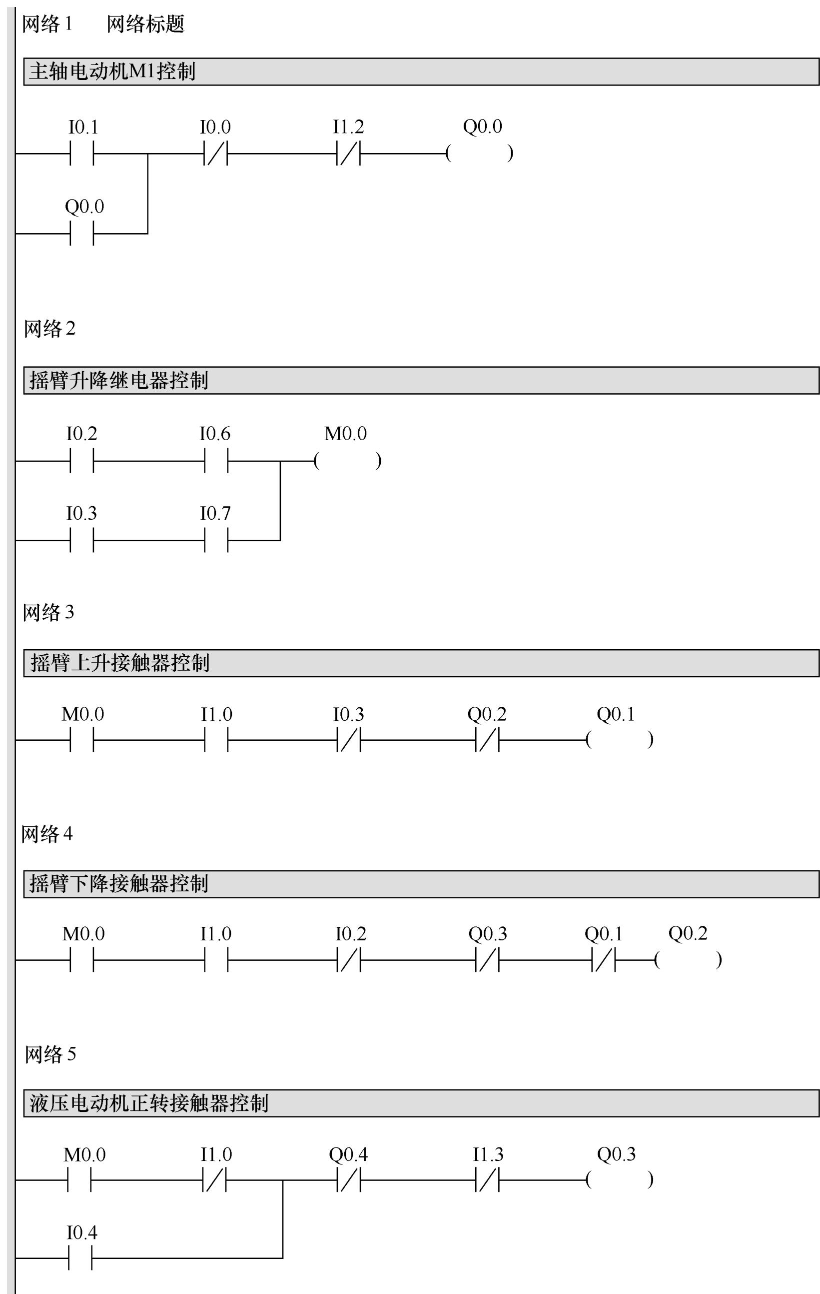


图 7.3-5　Z3040 钻床的 PLC 梯形图程序

3）现场调试中，接入实际的信号和负载时，应充分考虑各种可能的情况，做到认真、仔细、全面地完成现场调试。

4）注意人身和设备的安全。

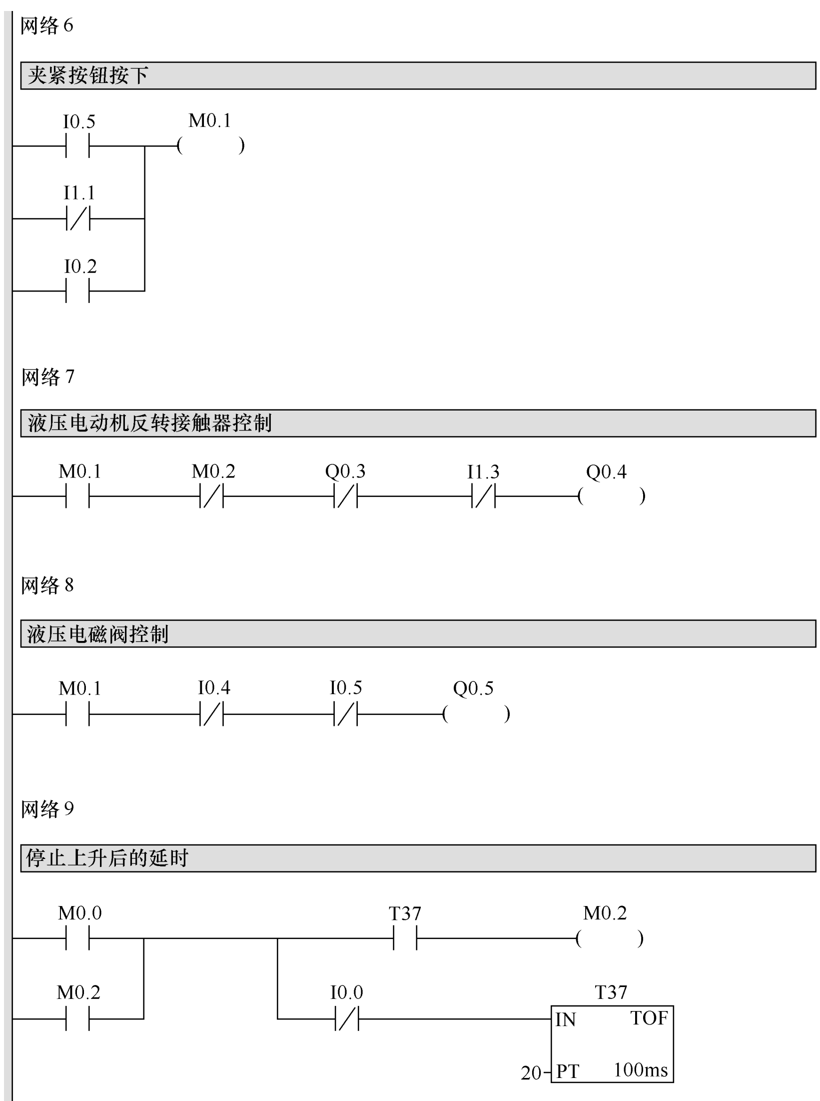


图 7.3-5　Z3040 钻床的 PLC 梯形图程序（续）

实例 4　M7120 平面磨床的 PLC 控制

平面磨床是用砂轮磨削加工各种零件平面的机床。M7120 平面磨床是平面磨床中使用较为普遍的一种，它的磨削精度高和表面较光洁，操作方便，适于磨削精密零件和各种工具。M7120 平面磨床主要由床身、工作台、电磁吸盘、砂轮箱、滑座、立柱等部分组成，如图 7.4-1 所示。

1. M7120 平面磨床的控制要求

（1）M7120 平面磨床控制系统的电力拖动形式

M7120 平面磨床采用分散拖动，共有四台电动机，即液压泵电动机、砂轮电动机、砂轮升降电动机和冷却泵电动机，全部采用普通笼型交流电动机。磨床的砂轮、砂轮升降和冷却

泵不要求调速，工作台往返运动是靠液压传动装置进行的，采用液压无级调速，运行较平稳，换向是通过工作台上的撞块碰撞床身上的液压换向开关来实现的。

（2）M7120 平面磨床控制系统的控制要求

1）砂轮电动机、液压泵电动机和冷却泵电动机只要求单方向旋转，因容量不大，故采用直接起动。

2）砂轮升降电动机要求能正反转。

3）冷却泵电动机要求在砂轮电动机运转后才能起动。

4）电磁吸盘需有去磁控制环节。

5）应具有完善的保护环节，如电动机的短路保护、过载保护、零电压保护及电磁吸盘的欠电压保护等。

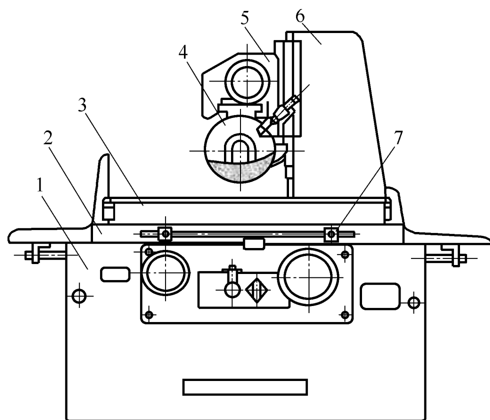


图 7.4-1　M7120 平面磨床外形
1—床身　2—工作台　3—电磁吸盘　4—砂轮箱
5—滑座　6—立柱　7—撞块

6）有必要的信号指示和局部照明。

2. M7120 平面磨床电气控制电路分析

图 7.4-2 为 M7120 平面磨床的电气控制电路图。

（1）主电路分析

主电路中共有四台电动机，M1 是液压泵电动机，实现工作台的往复运动；M2 是砂轮电动机，带动砂轮旋转磨削加工工件；M3 是冷却泵电动机，为砂轮磨削工件时输送冷却液；M4 是砂轮升降电动机，用以调整砂轮与工件的位置。其中砂轮升降电动机 M4 可正反转。四台电动机的工作要求是 M1、M2 和 M3 只需正转控制，M4 需要正反转控制，冷却泵电动机 M3 却需要在 M2 运转后才能运转。四台电动机具有短路、欠电压和失电压保护，分别由熔断器 FU1 和接触器 KM1、KM2、KM3 和 KM4 来执行，除 M4 之外，其余三台电动机分别由热继电器 FR1、FR2 和 FR3 进行过载保护。

（2）控制电路分析

当电源电压正常时，合上电源总开关 QS，位于 7 区的电压继电器 KUV 的常开触点闭合，便可进行操作。

1）液压泵电动机 M1 的控制。

起动过程：按下 SB3→KM1 得电→M1 起动。

停止过程：按下 SB2→KM1 失电→M1 停转。

运行过程中若 M1 过载，则 FR1 常闭触点分断，M1 停转，起到过载保护作用。

2）砂轮电动机 M2 的控制。

起动过程：按下 SB5→KM2 得电→M2 起动。

停止过程：按下 SB4→KM2 失电→M2 停转。

3）冷却液泵电动机 M3 的控制。冷却泵电动机 M3 通过接触器 KM2 控制，因此 M3 与砂轮电动机 M2 是联动控制。

按下 SB5 时 M3 与 M2 同时起动，按下 SB4 时 M3 与 M2 同时停止。

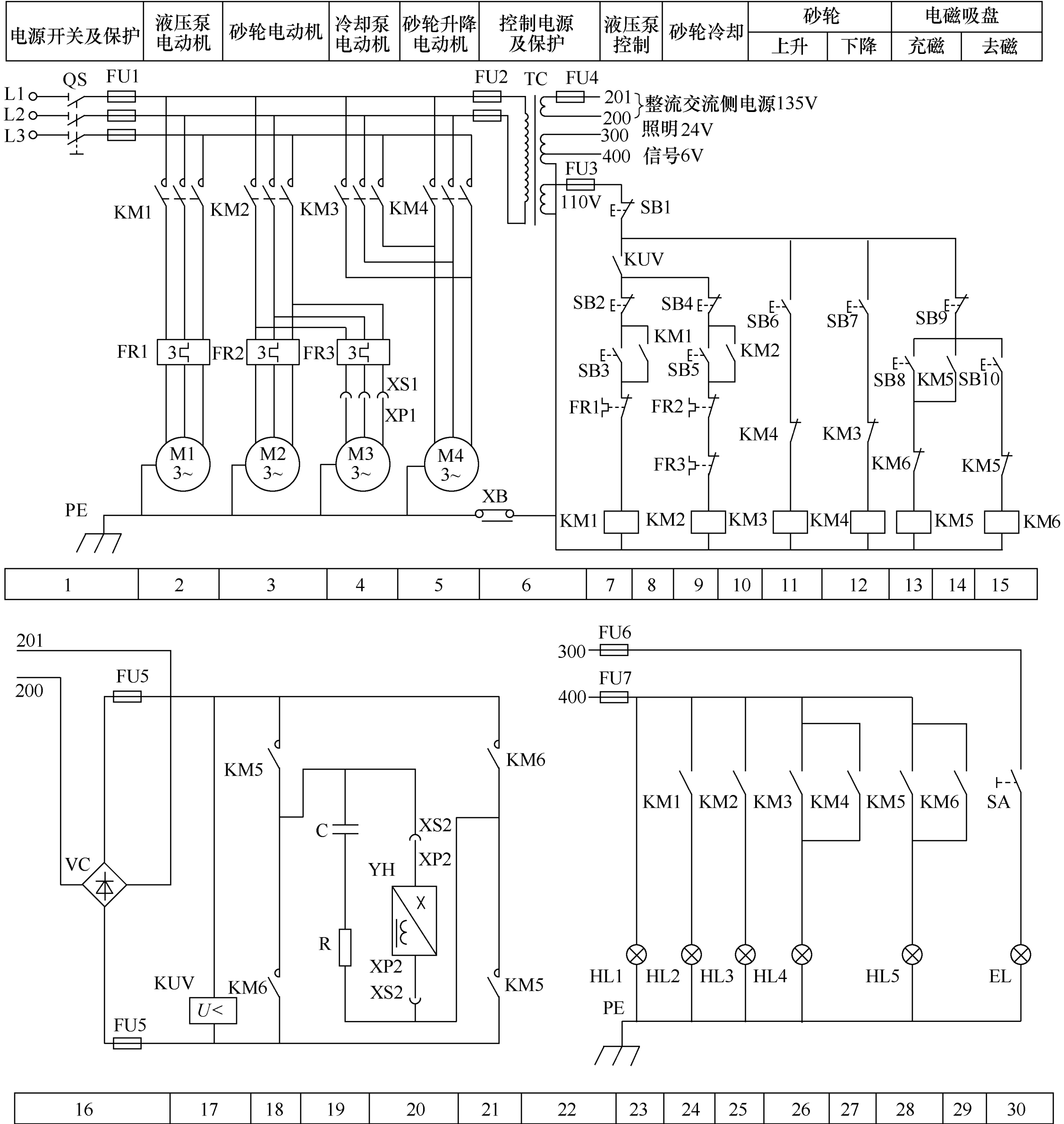


图 7.4-2　M7120 平面磨床电路图

FR2 与 FR3 的常闭触点串联在 KM2 线圈回路中。M2、M3 中任一台过载时，相应的热继电器动作，都将使 KM2 线圈失电，M2、M3 同时停止。

4）砂轮升降电动机 M4 的控制。砂轮升降电动机采用接触器联锁的点动正反转控制。

砂轮上升控制过程：按下 SB6→KM3 得电→M4 起动正转；当砂轮上升到预定位置时，松开 SB6→KM3 失电→M4 停转。

砂轮下降控制过程：按下 SB7→KM4 得电→M4 起动反转；当砂轮下降到预定位置时，松开 SB7→KM4 失电→M4 停转。

5）电磁工作台的控制。电磁工作台又称电磁吸盘，它是固定加工工件的一种夹具。

它利用通电线圈产生磁场的特性吸牢铁磁性材料的工件，便于磨削加工。电磁吸盘的内部装有凸起的磁极，磁极上绕有线圈。吸盘的面板也用钢板制成，在面板和磁极之间填有绝磁材料。当吸盘内的磁极线圈通以直流电时，磁极和面板之间形成两个磁极，既 N 极和 S 极。当工件放在两个磁极中间时，使磁路构成闭合回路，因此就将工件牢固地吸住。

① 电磁吸盘的组成：工作电路包括整流、控制和保护三个部分。整流部分由整流变压器和桥式整流器 VC 组成，输出 110V 直流电压。

② 电磁吸盘充磁的控制过程：按下 SB8→KM5 得电（自锁）→YH 充磁。

③ 电磁吸盘的退磁控制过程：工件加工完毕需取下时，先按下 SB9，切断电磁吸盘的电源，由于吸盘和工件都有剩磁，所以必须对吸盘和工件进行退磁。退磁过程为按下 SB10→KM6 得电→YH 退磁，此时电磁吸盘线圈通入反向的电流，以消除剩磁。由于去磁时间太长会使工件和吸盘反向磁化，因此去磁采用点动控制。松开 SB10 则去磁结束。

（3）辅助电路分析

辅助电路主要是信号指示和局部照明电路。其中，EL 为局部照明灯，由变压器 TC 供电，工作电压为 24V，由手动开关 SA 控制；信号灯也由 TC 供电，工作电压为 6V。HL1 为电源指示灯，HL2 为 M1 运转指示灯，HL3 为 M2 运转指示灯，HL4 为 M4 运转指示灯，HL5 为电磁吸盘工作指示灯。

3. 用 PLC 改造 M7120 平面磨床

（1）PLC 控制系统的主电路接线图

图 7.4-3 为 M7120 平面磨床 PLC 改造的主电路图。

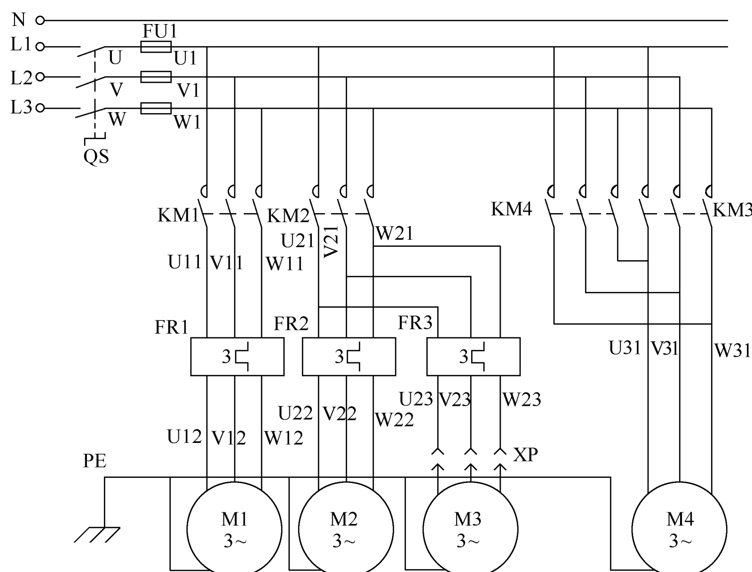


图 7.4-3 M7120 平面磨床 PLC 改造的主电路图

（2）PLC 控制系统的 I/O 接线

1）分配 PLC 的 I/O 地址通道。根据控制要求，首先确定 I/O 的个数，进行 I/O 的分配。本实例需要 13 个输入点，6 个输出点，见表 7.4-1。

表 7.4-1　PLC 的 I/O 地址通道分配

名　　称	输入元件	输　入　点	名　　称	输出元件	输　出　点
电压继电器	KUV	I0.0	液压泵电动机接触器	KM1	Q0.0
总停按钮	SB1	I0.1	砂轮电动机接触器 + 冷却泵电动机	KM2	Q0.1
液压泵电动机 M1 起动按钮	SB3	I0.2	砂轮上升接触器	KM3	Q0.2
液压泵电动机 M1 停止按钮	SB2	I0.3	砂轮下降接触器	KM4	Q0.3
砂轮电动机 M2 起动按钮	SB5	I0.4	电磁吸盘充磁接触器	KM5	Q0.4
砂轮电动机 M2 停止按钮	SB4	I0.5	电磁吸盘去磁接触器	KM6	Q0.5
砂轮升降电动机 M4 上升按钮	SB6	I0.6			
砂轮升降电动机 M4 下降按钮	SB7	I0.7			
电磁吸盘充磁按钮	SB9	I1.0			
电磁吸盘停止充磁按钮	SB8	I1.1			
电磁吸盘去磁按钮	SB10	I1.2			
液压泵电动机 M1 热继电器	FR1	I1.3			
砂轮电动机 M2 热继电器、冷却泵电动机 M3 热继电器	FR2、FR3	I1.4			

2）PLC 控制系统的 I/O 接线图。根据控制要求，设计并绘制 PLC 系统的 I/O 接线图，如图 7.4-4 所示。

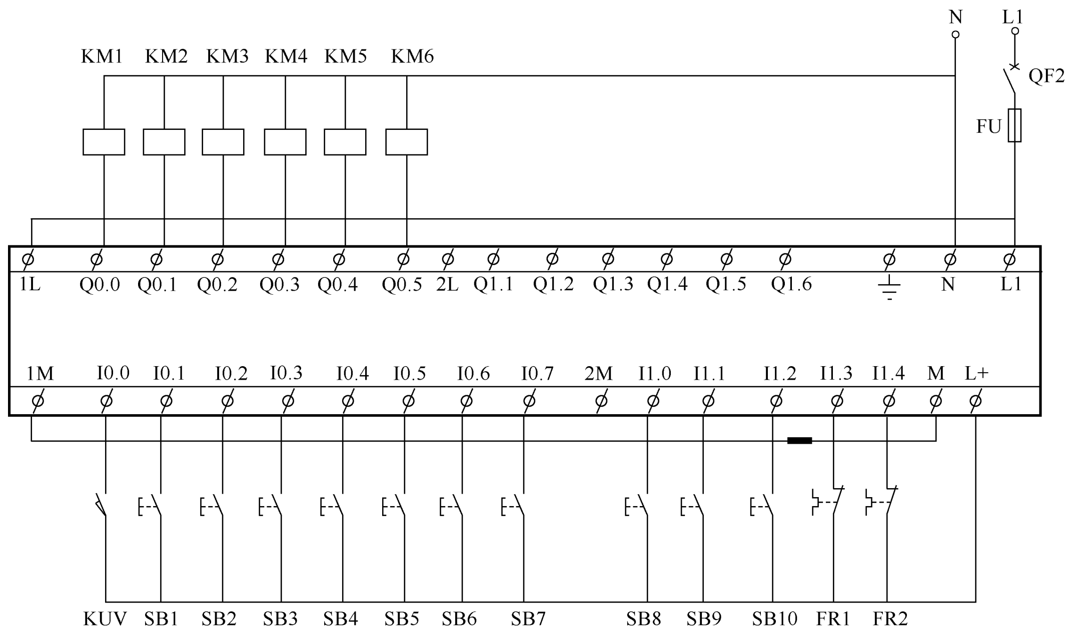


图 7.4-4　PLC 控制系统的 I/O 接线图

（3）安装与接线

1）将所有元件装在一块配电板上，做到布局合理、安装牢固、符合安装工艺规范。

2）根据接线原理图配线，做到接线正确、牢固、美观。

3）I/O 线和动力线应分开走线，并保持距离。数字量信号一般采用普通电缆就可以；

模拟信号线和高速信号线应采用屏蔽电缆，并做好接地工作。

4）安装 PLC 应远离强干扰源，并可靠接地，最好和强电的接地装置分开，接地线的截面积应大于 $2mm^2$，接地点与 PLC 的距离应小于 50cm。

（4）程序设计

图 7.4-5 为 M7120 平面磨床的 PLC 梯形图。

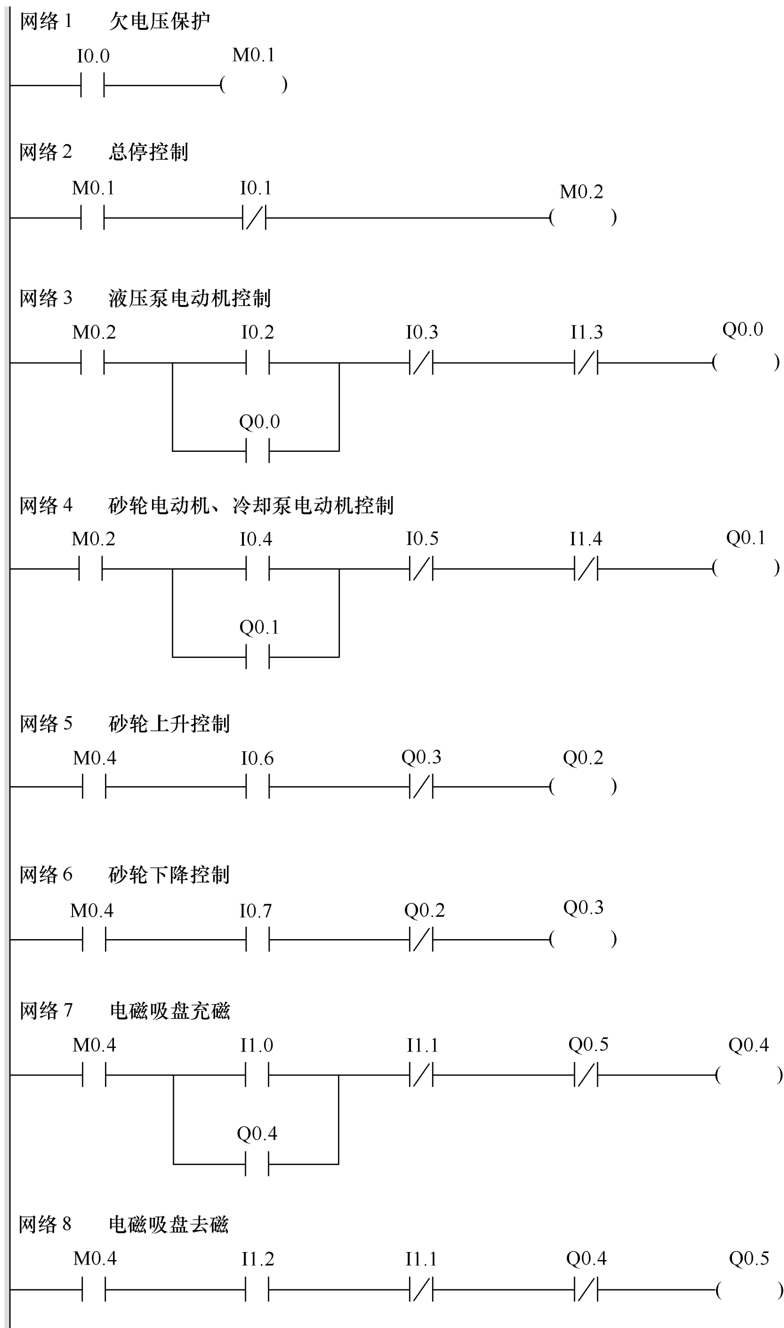


图 7.4-5 M7120 平面磨床的 PLC 梯形图

（5）程序输入与调试

熟练地操作编程软件，能正确将编制的程序输入 PLC；按照被控设备的要求进行调试、修改，达到设计要求。

1）通电前使用万用表检查电路的正确性，确保通电成功。

2）调试程序时先对程序进行模拟调试，对系统各种工作要求和方式都要逐一检查，不能遗漏，直到符合控制要求。

3）现场调试中，接入实际的信号和负载时，应充分考虑各种可能的情况，做到认真、仔细、全面地完成现场调试。

4）注意人身和设备的安全。




附　　录

附录 A　S7-200 的 SIMATIC 指令集

布尔指令	
LD　N LDI　N LDN　N LDNI　N	装载（开始的常开触点） 立即装载 取反后装载（开始的常闭触点） 取反后立即装载
A　N AI　N AN　N ANI　N	与（串联的常开触点） 立即与 取反后与（串联的常开触点） 取反后立即与
O　N OI　N ON　N ONI　N	或（并联的常开触点） 立即或 取反后或（并联的常开触点） 取反后立即与
LDBx　N1，N2	装载字节比较结果 N1（x：<，< =，=，> =，>，< > =）N2
ABx　N1，N2	与字节比较结果 N1（x：<，< =，=，> =，>，< > =）N2
OBx　N1，N2	或字节比较结果 N1（x：<，< =，=，> =，>，< > =）N2
LDWx　N1，N2	装载字比较结果 N1（x：<，< =，=，> =，>，< > =）N2
AWx　N1，N2	与字比较结果 N1（x：<，< =，=，> =，>，< > =）N2
OWx　N1，N2	或字比较结果 N1（x：<，< =，=，> =，>，< > =）N2
LDDx　N1，N2	装载双字比较结果 N1（x：<，< =，=，> =，>，< > =）N2
ADx　N1，N2	与双字比较结果 N1（x：<，< =，=，> =，>，< > =）N2
ODx　N1，N2	或双字比较结果 N1（x：<，< =，=，> =，>，< > =）N2
LDRx　N1，N2	装载实数比较结果 N1（x：<，< =，=，> =，>，< > =）N2
ARx　N1，N2	与实数比较结果 N1（x：<，< =，=，> =，>，< > =）N2
ORx　N1，N2	或实数比较结果 N1（x：<，< =，=，> =，>，< > =）N2
LDSx　N1，N2	装载字符串比较结果 N1（x：=，< >）N2
ASx　N1，N2	与字符串比较结果 N1（x：=，< >）N2

（续）

	布尔指令	
OSx	N1，N2	或字符串比较结果 N1（x：=，< >）N2
NOT		栈顶值取反
EU ED		上升沿检测 下降沿检测
= =I	N N	赋值（线圈） 立即赋值
S R SI RI	S_BIT，N S_BIT，N S_BIT，N S_BIT，N	置位一个区域 复位一个区域 立即置位一个区域 立即复位一个区域
	传送、移位、循环和填充指令	
MOVB MOVW MOVD MOVR BIR BIW	IN，OUT IN，OUT IN，OUT IN，OUT IN，OUT IN，OUT	字节传送 字传送 双字传送 实数传送 立即读取物理输入字节 立即写物理输出字节
BMB BMW BMD	IN，OUT，N IN，OUT，N IN，OUT，N	字节块传送 字块传送 双字块传送
SWAP	IN	交换字节
SHRB	DATA，S_BIT，N	移位寄存器
SRB SRW SRD	OUT，N OUT，N OUT，N	字节右移 N 位 字右移 N 位 双字右移 N 位
SLB SLW SLD	OUT，N OUT，N OUT，N	字节左移 N 位 字左移 N 位 双字左移 N 位
RRB RRW RRD	OUT，N OUT，N OUT，N	字节右移 N 位 字右移 N 位 双字右移 N 位
RLB RLW RLD	OUT，N OUT，N OUT，N	字节左移 N 位 字左移 N 位 双字左移 N 位
FILL	IN，OUT，N	用指定的元素填充存储器空间
	字符串指令	
SLEN	IN，OUT	字符串长度
SCPY	IN，OUT	字符串复制

（续）

字符串指令	
SCAT　IN，OUT	字符串连接
SSCPY　IN，INDX，N，OUT	从字符串中复制子字符串
SFND　IN1，IN2，OUT	字符串搜索
CFND　IN1，IN2，OUT	字符搜索
逻辑操作指令	
ALD OLD	电路块串联 电路块并联
LPS LRD LPP LDS	入栈 读栈 出栈 装载堆栈
AENO	对 ENO 进行与操作
ANDB　IN1，OUT ANDW　IN1，OUT ANDD　IN1，OUT	字节逻辑与 字逻辑与 双字逻辑与
ORB　IN1，OUT ORW　IN1，OUT ORD　IN1，OUT	字节逻辑或 字逻辑或 双字逻辑或
XORB　IN1，OUT XORW　IN1，OUT XORD　IN1，OUT	字节逻辑异或 字逻辑异或 双字逻辑异或
INVB　OUT INVW　OUT INVD　OUT	字节取反（1 的补码） 字取反 双字取反
表功能指令、BCD 码/数据类型/ASCII 码转换指令、译/编码指令、段译码指令	
ATT　TABLE，DATA LIFO　TABLE，DATA FIFO　TABLE，DATA	把数据加到表中 从表中取数据，后入先出 从表中取数据，先入先出
FND =　TBL，PATRN，INDX FND < >　TBL，PATRN，INDX FND <　TBL，PATRN，INDX FND >　TBL，PATRN，INDX	在表中查找符合比较条件的数据
BCDI　OUT IBCD　OUT　BCD	码转换成整数 整数转换成 BCD 码
BTI　IN，OUT IBT　IN，OUT ITD　IN，OUT TDI　IN，OUT	字节转换成整数 整数转换成字节 整数转换成双整数 双整数转换成整数

（续）

表功能指令、BCD码/数据类型/ASCII码转换指令、译/编码指令、段译码指令		
DTR TRUNC ROUND	IN，OUT IN，OUT IN，OUT	双整数转换成实数 实数四舍五入为双整数 实数截位取整为双整数
ATH HTA ITA DTA RTA	IN，OUT，LEN IN，OUT，LEN IN，OUT，FMT IN，OUT，FMT IN，OUT，FMT	ASCII码→十六进制数 十六进制数→ASCII码 整数→ASCII码 双整数→ASCII码 实数→ASCII码
DECO ENCO	IN，OUT IN，OUT	译码 编码
SEG	IN，OUT	七段译码
中断指令		
CRETI		从中断程序有条件返回
ENI DISI		允许中断 禁止中断
ATCH DTCH	INT，EVENT EVENT	给事件分配中断程序 解除中断事件
通信指令		
XMT RCV	TABLE，PORT TABLE，PORT	自由端口发送 自由端口接收
NETR NETW	TABLE，PORT TABLE，PORT	网络读 网络写
GPA SPA	ADDR，PORT ADDR，PORT	获取端口地址 设置端口地址
高速计数器指令		
HDEF	HSC，MODE	定义高速计数器模式
HSC	N	激活高速计数器
PLS	X	脉冲输出
算术运算、数学功能、加1/减1、PID指令		
+I +D +R	IN1，OUT IN1，OUT IN1，OUT	整数、双整数或实数加法 IN1 + OUT = OUT
-I -D -R	IN1，OUT IN1，OUT IN1，OUT	整数、双整数或实数减法 OUT - IN1 = OUT

（续）

算术运算、数学功能、加 1/减 1、PID 指令	
MUL　IN1，OUT ＊R　IN1，OUT ＊I　IN1，OUT ＊D　IN1，OUT	整数乘整数得双整数 实数、整数或双整数乘法 IN1 × OUT = OUT
MUL　IN1，OUT /R　IN1，OUT /I　IN1，OUT /D　IN1，OUT	整数除整数得双整数 实数、整数或双整数除法 OUT/IN1 = OUT
SQRT　IN，OUT	二次方根
LN　IN，OUT	自然对数
LXP　IN，OUT	自然指数
SIN　IN，OUT	正弦
COS　IN，OUT	余弦
TAN　IN，OUT	正切
INCB　OUT	字节加 1
INCW　OUT	字加 1
INCD　OUT	双字加 1
DECB　OUT DECW　OUT DECD　OUT	字节减 1 字减 1 双字减 1
PID　Table，Loop	PID 回路
定时器和计数器指令	
TON　Txxx，PT TOF　Txxx，PT TONR　Txxx，PT	通电延时定时器 断电延时定时器 保持型通电延时定时器
CTU　Txxx，PV CTD　Txxx，PV CTUD　Txxx，PV	增计数器 减计数器 增/减计数器
实时时钟指令	
TODR　T TODW　T	读实时时钟 写实时时钟
程序控制指令	
END	程序的条件结束
STOP	切换到 STOP 模式
WDR	看门狗复位（300ms）
JMP　N	跳到指定的标号
LBL　N	定义一个跳转的标号

（续）

程序控制指令	
CALL　N（N1，…） CRET	调用子程序，可以有 16 个可选参数 从子程序条件返回
FOR　INDX，INIT，FINAL N EXT	For/Next 循环
LSCR　N SCRT　N SCRE	顺控继电器段的启动 顺控继电器段的转换 顺控断电器段的结束

附录 B　S7-200 的特殊存储器（SM）标志位

特殊内存位提供各种状态和控制功能，也是一种在 S7-200 和用户程序之间通信的方式。特殊内存位可以被用作位、字节、字或双字。

1. SMB0：状态位

SMB0 包含八个状态位，它们在每个扫描循环的结束由 S7-200 更新，见表 B-1。

表 B-1　特殊内存字节 SMB0（SM0.0～SM0.7）

SM 位	说　明
SM0.0	此位始终接通
SM0.1	此位在首次扫描周期接通。一个用途是调用初始化子例行程序
SM0.2	如果保留性数据丢失，此位在一个扫描循环内变为接通。此位可以用作错误内存位或用作调用特殊启动顺序的机制
SM0.3	当从上电条件进入 RUN（运行）模式时，此位变为一个扫描循环而接通。此位可以用作在开始操作前提供机器预热时间
SM0.4	此位提供时钟脉冲，对于 1min 的工作循环时间，30s 接通，30s 断开。它提供容易使用的延迟，或者 1min 时钟脉冲
SM0.5	此位提供时钟脉冲，对于 1s 的工作循环时间，0.5s 接通，0.5s 断开。它提供容易使用的延迟，或者 1s 时钟脉冲
SM0.6	此位是扫描循环脉冲，在一个扫描循环接通，然后在下一个扫描循环断开。此位可以用作扫描计数器输入
SM0.7	此位反映了模式开关的位置（断开是 TERM 位置，接通是 RUN 位置）。如果当开关在 RUN 位置时使用此位启用自由端口模式，与编程设备的正常通信可以通过切换到 TERM 位置来启用

2. SMB1：状态位

SMB1 包含各种电位出错指示器，见表 B-2。这些位在执行时由指令置位和重设。

表 B-2　特殊内存字节 SMB1（SM1.0～SM1.7）

SM 位	说　明
SM1.0	当操作结果为零时，此位通过执行某些指令而接通
SM1.1	当引起溢出或当检测到非法的数字值时，此位通过执行某些指令而接通
SM1.2	当通过算术运算产生负结果时，此位接通
SM1.3	当尝试除以零时，此位接通
SM1.4	当“添加到表格”指令试图填满表格时，此位接通
SM1.5	当 LIFO 或 FIFO 指令尝试从空表读时，此位接通
SM1.6	当进行尝试转换非 BCD 码数值到二进制时，此位接通
SM1.7	当 ASCII 数值无法转换为有效的十六进制数值时，此位接通

3. SMB2：自由端口接收字符

SMB2 是自由端口接收字符缓冲区。如表 B-3 所述，在自由端口模式下接收的每个字符放在此位置中，以便梯形程序存取。

表 B-3　特殊内存字节 SMB2

SM 位	说　明
SMB2	在自由口通信方式下，该字符存储从口 0 或口 1 接收到的每一个字符

注：SMB2 和 SMB3 在端口 0 和端口 1 之间共享。当接收端口 0 上的字符导致执行附加在那个事件（中断事件 8）的中断例行程序时，SMB2 包含端口 0 上接收的字符，而 SMB3 包含该字符的奇偶校验状态。当接收端口 1 上的字符导致执行附加在那个事件（中断事件 25）的中断例行程序时，SMB2 包含端口 1 上接收的字符，而 SMB3 包含该字符的奇偶校验状态。

4. SMB3：自由端口奇偶校验错误

SMB3 用于自由端口模式并包含奇偶校验错误位，当在接收的字符上检测到奇偶校验出错时，该位就被置位。当检测到奇偶校验出错时，SM3.0 接通，使用此位放弃消息，见表 B-4。

表 B-4　特殊内存字节 SMB3（SM3.0～SM3.7）

SM 位	说　明
SM3.0	来自端口 0 或端口 1 的奇偶校验错误（0＝无错，1＝检测到错误）
SM3.1～SM3.7	保留

5. SMB4：队列溢出

如表 B-5 所述，SMB4 包含中断队列溢出位、中断是否允许标志位及发送空闲位。队列溢出表明要么是中断发生的频率高于 CPU，要么是中断已经被全局中断禁止指令所禁止。

表 B-5　特殊内存字节 SMB4（SM4.0～SM4.7）

SM 位	说　明
SM4.0	当通信中断队列溢出时，此位接通
SM4.1	当输入中断队列溢出时，此位接通

（续）

SM 位	说　　明
SM4.2	当定时中断队列溢出时，此位接通
SM4.3	当检测到运行系统程序问题时，此位接通
SM4.4	此位反映全局中断启用状态。当中断启用时，此位接通
SM4.5	当发送器闲置时（端口 0），此位接通
SM4.6	当发送器闲置时（端口 1），此位接通
SM4.7	当发生强制时，此位接通

注：在中断例行程序中只使用状态位 SM4.0、SM4.1 和 SM4.2。当队列被清空时，这些状态位重设，并且控制返回到主程序。

6. SMB5：I/O 状态

SMB5 是特殊内存字节，它包含在 I/O 系统中检出的出错条件的状态位，见表 B-6。这些位提供检测出的 I/O 错误概况。

表 B-6　特殊内存字节 SMB5（SM5.0 ~ SM5.7）

SM 位	说　　明
SM5.0	如果显示任何 I/O 错误，此位接通
SM5.1	如果太多的数字 I/O 点连接到 I/O 总线，此位接通
SM5.2	如果太多的模拟 I/O 点连接到 I/O 总线，此位接通
SM5.3	如果太多的智能 I/O 模块连接到 I/O 总线，此位接通
SM5.4 ~ SM5.7	保留

7. SMB6：CPU 标识寄存器

SMB6 是 S7-200 CPU 的标识寄存器，见表 B-7。SM6.4 ~ SM6.7 识别 S7-200 CPU 的型号，SM6.0 ~ SM6.3 保留作为将来使用。

表 B-7　特殊内存字节 SMB6（SM6.0 ~ SM6.7）

SM 位	说　　明
格式	MSB 7 … LSB 0 × × × × r r r r　CPU 识别（ID）寄存器
SM6.0 ~ SM6.3	保留
SM6.4 ~ SM6.7	xxxx = 0000 = CPU222 0010 = CPU224 0110 = CPU221 1001 = CPU226/CPU226XM

8. SMB7：保留

SMB7 保留作为将来使用。

9. SMB8 ~ SMB21：I/O 模块标识号和错误寄存器

SMB8 ~ SMB21 以字节对组织用于扩充模块 0 ~ 6。如表 B-8 所述，每个对的偶数字节是模块标识寄存器，这些字节识别模块类型、I/O 类型以及输入和输出的数目；每个对的奇数字节是模块错误寄存器，这些字节提供在 I/O 检测出的该模块的任何错误的指示。

表 B-8　特殊内存字节 SMB8 ~ SMB21

SM 位	说　明
格式	偶数字节：模块标识寄存器 MSB 7 … LSB 0 | m | t | t | a | i | i | q | q | m：模块存在　0 = 存在　1 = 不存在 tt：模块类型 00　非智能 I/O 模块 01　智能模块 10　保留 11　保留 a：I/O 类型　0 = 离散　1 = 模拟 ii：输入 00　无输入 01　2 AI 或 8 DI 10　4 AI 或 16 DI 11　8 AI 或 32 DI qq：输出 00　无输出 01　2 AQ 或 8 DQ 10　4 AQ 或 16 DQ 11　8 AQ 或 32 DQ 奇数字节：模块错误寄存器 MSB 7 … LSB 0 | c | 0 | 0 | b | r | p | f | t | c：组态错误　0 = 无错误 b：总线故障或奇偶校验错误　1 = 错误 r：超出范围错误 p：无用户电源错误 f：熔丝熔断错误 t：接线板松动错误
SMB8	模块 0 标识寄存器
SMB9	模块 0 错误寄存器
SMB10	模块 1 标识寄存器
SMB11	模块 1 错误寄存器
SMB12	模块 2 标识寄存器
SMB13	模块 2 错误寄存器
SMB14	模块 3 标识寄存器
SMB15	模块 3 错误寄存器
SMB16	模块 4 标识寄存器
SMB17	模块 4 错误寄存器
SMB18	模块 5 标识寄存器
SMB19	模块 5 错误寄存器
SMB20	模块 6 标识寄存器
SMB21	模块 6 错误寄存器

10. SMW22 ~ SMW26：扫描时间

如表 B-9 所述，SMW22、SMW24 和 SMW26 提供扫描时间信息：最小扫描时间、最大扫描时间和最后扫描时间（以 ms 为单位）。

表 B-9 特殊内存字 SMW22 ~ SMW26

SM 位	说 明
SMW22	最后扫描循环的扫描时间（以 ms 为单位）
SMW24	从进入 RUN（运行）模式开始记录的最小扫描时间（以 ms 为单位）
SMW26	从进入 RUN（运行）模式开始记录的最大扫描时间（以 ms 为单位）

11. SMB28 和 SMB29：模拟调整

SMB28 包含代表模拟电位器 0 位置的数字值，SMB29 包含代表模拟电位器 1 位置的数字值，见表 B-10。

表 B-10 特殊内存字节 SMB28 和 SMB29

SM 字节	说 明
SMB28	此字节存储以模拟电位器 0 输入的数值。在每次停止/运行扫描中，此数值更新一次
SMB29	此字节存储以模拟电位器 1 输入的数值。在每次停止/运行扫描中，此数值更新一次

12. SMB30 和 SMB130：自由端口控制寄存器

SMB30 控制端口 0 的自由端口通信，SMB130 控制端口 1 的自由端口通信。可以读出和写入至 SMB30 和 SMB130。这些字节为自由端口操作配置各自的通信端口，并提供自由端口或系统协议支持的选择，见表 B-11。

表 B-11 特殊内存字节 SMB30 和 SMB130

端 口 0	端 口 1	说 明
SMB30 的格式	SMB130 的格式	自由端口模式控制字节 MSB 7 … LSB 0 p p d b b b m m
SM30. 0 和 SM30. 1	SM130. 0 和 SM130. 1	mm：协议选择 00 = 点对点接口协议 （PPI/从站模式） 01 = 自由端口协议 10 = PPI/主站模式 11 = 保留（默认设置为 PPI/从站模式） 注意：当选择代码 mm = 10（PPI/主站）时，S7-200 将成为网络上的主站，允许执行 NETR 和 NETW 指令。在 PPI 模式下忽略 2 ~ 7 位
SM30. 2 ~ SM30. 4	SM130. 2 ~ SM130. 4	bbb：自由端口波特率 000 = 38400 波特，100 = 2400 波特 001 = 19200 波特，101 = 1200 波特 010 = 9600 波特，110 = 115200 波特 011 = 4800 波特，111 = 57600 波特

（续）

端　口　0	端　口　1	说　　明
SM30.5	SM130.5	d：每个字符的数据位 0 = 每个字符 8 位 1 = 每个字符 7 位
SM30.6 和 SM30.7	SM130.6 和 SM130.7	pp：奇偶校验选择 00 = 无奇偶校验，10 = 无奇偶校验 01 = 偶校验，11 = 奇校验

13. SMB31 和 SMW32：永久存储器（EEPROM）写控制

在用户程序的控制下，可以把 V 存储器中的数据存入永久存储器，亦称非易失存储器。先把被存数据的地址存入 SMW32 中，然后把存入命令存入 SMB31 中。一旦用户发出存储命令，则直到 CPU 完成存储操作 SM31.7 被置 0 之前，都不可以改变 V 存储器的值。

在每次扫描周期末尾，CPU 检查是否有向永久存储器中存数据的命令。如果有，则将该数据存入永久存储器中。

如表 B-12 所示，SMB31 定义了存入永久存储器的数据大小，且提供了初始化存储操作的命令；SMW32 提供了被存数据在 V 存储器中的起始地址。

表 B-12　特殊存储器字节 SMB31 和特殊存储器字 SMW32

SM 位	说　　明
格式	SMB31：软件命令 MSB 7 … LSB 0 c | 0 | 0 | 0 | 0 | 0 | s | s SMW32：V 存储器地址 MSB 15 … LSB 0 V 存储器地址
SM31.0 和 SM31.1	ss：数据大小 00 = 字节，10 = 字，01 = 字节，11 = 双字
SM31.7	c：保存至永久存储器 0 = 无执行保存操作的请求 1 = 用户程序请求保存数据 每次存储操作完成后，S7-200 复位该位
SMW32	SMW32 中是所存数据的 V 存储器地址，该值是相对于 V0 的偏移量。当执行存储命令时，把该数据存到永久存储器中相应的位置

14. SMB34 和 SMB35：用于定时中断的时间间隔寄存器

如表 B-13 所示，SMB34 和 SMB35 分别定义了定时中断 0 和 1 的时间间隔，可以在 1 ~ 255ms 之间以 1ms 为增量进行设定。如果相应的定时中断事件被连接到一个中断服务程序，S7-200 就会获取该时间间隔值。若要改变该时间间隔，用户必须把定时中断事件再分配给

同一或另一中断程序，也可以通过中断分离来终止定时中断事件。

表 B-13　特殊存储器字节 SMB34 和 SMB35

SM 位	说　明
SMB34	定义定时中断 0 的时间间隔（1～255ms，以 1ms 为增量）
SMB35	定义定时中断 1 的时间间隔（1～255ms，以 1ms 为增量）

15. SMB36～SMD65：HC0、HC1 和 HC2 寄存器

如表 B-14 所示，SMB36～SMD65 用于监视和控制高速计数器 HC0、HC1 和 HC2 的操作。

表 B-14　特殊存储器字节 SMB36～SMD65

SM 位	说　明
SM36.0～SM36.4	保留
SM36.5	HC0 当前计数方向状态位：1＝增计数
SM36.6	HC0 当前值等于预设值状态位：1＝相等
SM36.7	HC0 当前值大于预设值状态位：1＝大于
SM37.0	复位的有效电平控制位：0＝复位为高电平有效，1＝复位为低电平有效
SM37.1	保留
SM37.2	正交计数器的计数速率选择：0＝4×计数速率；1＝1×计数速率
SM37.3	HC0 方向控制位：1＝增计数
SM37.4	HC0 更新方向：1＝更新方向
SM37.5	HC0 更新预设值：1＝将新预设值写入 HC0 预设值
SM37.6	HC0 更新当前值：1＝将新当前值写入 HC0 当前值
SM37.7	HC0 启用位：1＝启用
SMD38 SMD39 SMD40 SMD41	HC0 新的当前值 SMD38 是最高有效字节，SMD41 是最低有效字节
SMD42 SMD43 SMD44 SMD45	HC0 新的预置值 SMD42 是最高有效字节，SMD45 是最低有效字节
SM46.0～SM46.4	保留
SM46.5	HC1 当前计数方向状态位：1＝增计数
SM46.6	HC1 当前值等于预设值状态位：1＝等于
SM46.7	HC1 当前值大于预设值状态位：1＝大于
SM47.0	HC1 复位的有效电平控制位：0＝高电平有效，1＝低电平有效
SM47.1	HC1 启动的有效电平控制位：0＝高电平有效，1＝低电平有效
SM47.2	HC1 正交计数器速率选择：0＝4×速率，1＝1×速率
SM47.3	HC1 方向控制位：1＝增计数
SM47.4	HC1 更新方向：1＝更新方向

（续）

SM 位	说　明
SM47.5	HC1 更新预设值：1＝将新预设值写入 HC1 预设值
SM47.6	HC1 更新当前值：1＝将新当前值写入 HC1 当前值
SM47.7	HC1 启用位：1＝启用
SMD48 SMD49 SMD50 SMD51	HC1 新的当前值 SMD48 是最高有效字节，SMD51 是最低有效字节
SMD52 SMD53 SMD54 SMD55	HC1 新的预置值 SMD52 是最高有效字节，SMD55 是最低有效字节
SM56.0～SM56.4	保留
SM56.5	HC2 当前计数方向状态位：1＝增计数
SM56.6	HC2 当前值等于预设值状态位：1＝等于
SM56.7	HC2 当前值大于预设值状态位：1＝大于
SM57.0	HC2 复位的有效电平控制位：0＝高电平有效，1＝低电平有效
SM57.1	HC2 启动的有效电平控制位：0＝高电平有效，1＝低电平有效
SM57.2	HC2 正交计数器速率选择：0＝4×速率，1＝1×速率
SM57.3	HC2 方向控制位：1＝增计数
SM57.4	HC2 更新方向：1＝更新方向
SM57.5	HC2 更新预设值：1＝将新设置值写入 HC2 预设值
SM57.6	HC2 更新当前值：1＝将新当前值写入 HC2 当前值
SM57.7	HC2 启用位：1＝启用
SMD58 SMD59 SMD60 SMD61	HC2 新的当前值 SMD58 是最高有效字节，SMD61 是最低有效字节
SMD62 SMD63 SMD64 SMD65	HC2 新的预置值 SMD62 是最高有效字节，SMD65 是最低有效字节

16. SMB66～SMB85：PTO/PWM 寄存器

如表 B-15 所示，SMB66～SMB85 用于监视和控制脉冲串输出（PTO）和脉宽调制（PWM）功能。

表 B-15　特殊存储器字节 SMB66～SMB85

SM 位	说　明
SM66.0～SM66.3	保留
SM66.4	PTO0 包络被中止：0＝无错，1＝因增量计算错误而被中止

（续）

SM 位	说　　明
SM66.5	PTO0 包络被中止：0 = 不通过用户命令中止，1 = 通过用户命令中止
SM66.6	PTO0/PWM 管线溢出（在使用外部包络时由系统清除，否则必须由用户复位）：0 = 无溢出，1 = 管线溢出
SM66.7	PTO0 空闲位：0 = PTO 正在执行，1 = PTO 空闲
SM67.0	PTO0/PWM0 更新周期值：1 = 写入新周期
SM67.1	PWM0 更新脉宽值：1 = 写入新脉宽
SM67.2	PTO0 更新脉冲计数值：1 = 写入新脉冲计数
SM67.3	PTO0/PWM0 时间基准：0 = 1μs/刻度，1 = 1ms/刻度
SM67.4	同步更新 PWM0：0 = 异步更新，1 = 同步更新
SM67.5	PTO0 操作：0 = 单段操作（周期和脉冲计数存储在 SM 存储器中），1 = 多段操作（包络表存储在 V 存储器中）
SM67.6	PTO0/PWM0 模式选择：0 = PTO，1 = PWM
SM67.7	PTO0/PWM0 启用位：1 = 启用
SMW68 SMD69	PTO0/PWM0 周期（2 ~ 65535 个时间基准） SMD68 是最高有效字节，SMD69 是最低有效字节
SMW70 SMD71	PWM0 脉冲宽度值（0 ~ 65535 个） SMD70 是最高有效字节，SMD71 是最低有效字节
SMD72 SMD73 SMD74 SMD75	PTO0 脉冲计数值（$1 \sim 2^{32}-1$） SMD72 是最高有效字节，SMD75 是最低有效字节
SM76.0 ~ SM76.3	保留
SM76.4	PTO1 包络被中止：0 = 无错，1 = 因增量计算错误而被中止
SM76.5	PTO1 包络被中止：0 = 不通过用户命令中止，1 = 通过用户命令中止
SM76.6	PTO1/PWM 管线溢出（在使用外部包络时由系统清除，否则必须由用户复位）：0 = 无溢出，1 = 管线溢出
SM76.7	PTO1 空闲位：0 = PTO 正在执行，1 = PTO 空闲
SM77.0	PTO1/PWM1 更新周期值：1 = 写入新周期
SM77.1	PWM1 更新脉宽值：1 = 写入新脉宽
SM77.2	PTO1 更新脉冲计数值：1 = 写入新脉冲计数
SM77.3	PTO1/PWM1 时间基准：0 = 1μs/刻度，1 = 1ms/刻度
SM77.4	同步更新 PWM1：0 = 异步更新，1 = 同步更新
SM77.5	PTO1 操作：0 = 单段操作（周期和脉冲计数存储在 SM 存储器中），1 = 多段操作（包络表存储在 V 存储器中）
SM77.6	PTO1/PWM1 模式选择：0 = PTO，1 = PWM
SM77.7	PTO1/PWM1 启用位：1 = 启用

（续）

SM 位	说明
SMW78 SMW79	PTO1/PWM1 周期值（2～65535 个时间基准） SMD78 是最高有效字节，SMD79 是最低有效字节
SMW80 SMW81	PWM1 脉冲宽度值（0～65535 个时间基准） SMD80 是最高有效字节，SMD81 是最低有效字节
SMD82 SMW83 SMW84 SMW85	PTO1 脉冲计数值（$1 \sim 2^{32}-1$） SMD82 是最高有效字节，SMD85 是最低有效字节

17. SMB86～SMB94，SMB186～SMB194：接收消息控制

如表 B-16 所示，SMB86～SMB94 和 SMB186～SMB194 用于控制和读出接收消息指令的状态。

表 B-16 特殊存储器字节 SMB86～SMB94，SMB186～SMB194

端口 0	端口 1	说明
SMB86	SMB186	接收消息状态字节 MSB 7 … LSB 0 n | r | e | 0 | 0 | t | c | p n：1 = 接收消息通过用户禁用命令终止 r：1 = 接收消息被终止：输入参数出错或缺失启动或结束条件 e：1 = 结束字符已接收 t：1 = 接收消息被终止：定时器时间用完 c：1 = 接收消息被终止：达到最大字符计数 p：1 = 接收消息终止：校验错误
SMB87	SMB187	接收消息控制字节 MSB 7 … LSB 0 en | sc | ec | il | c/m | tmr | bk | 0 en：0 = 接收消息功能被禁用，1 = 允许接收消息功能 每次执行 RCV 指令时检查允许/禁止接收消息位 sc：0 = 忽略 SMB88 或 SMB188，1 = 使用 SMB88 或 SMB188 的值检测起始消息 ec：0 = 忽略 SMB89 或 SMB189，1 = 使用 SMB89 或 SMB189 的值检测结束消息 il：0 = 忽略 SMW90 或 SMW190，1 = 使用 SMW90 或 SMW190 的值检测空闲状态 c/m：0 = 定时器是字符间隔定时器，1 = 定时器是消息定时器 tmr：0 = 忽略 SMW92 或 SMW192，1 = 当 SMW92 或 SMW192 中的定时时间超出时终止接收 bk：0 = 忽略中断条件，1 = 用中断条件作为消息检测的开始
SMB88	SMB188	消息字符的开始
SMB89	SMB189	消息字符的结束

（续）

端　口　0	端　口　1	说　　明
SMW90 SMW91	SMW190 SMW191	空闲行时间段按毫秒设定。空闲行时间用完后接收的第一个字符是新消息的开始。SMD90（或 SMW190）是最高有效字节，SMD91（或 SMW191）是最低有效字节
SMW92 SMW93	SMW192 SMW93	字符间/消息间定时器超时值（用 ms 表示）。如果超过时间，就停止接收消息。SMD92（或 SMW192）是最高有效字节，SMD93（或 SMW193）是最低有效字节
SMB94	SMB194	要接收的最大字符数（1～255 字节） 注意：此范围必须设置为期望的最大缓冲区大小，即使在不使用字符计数消息终止功能时也是如此

18. SMW98：扩展 I/O 总线错误

如表 B-17 所示，SMW98 给出有关扩展 I/O 总线的错误数的信息。

表 B-17　特殊存储器字节 SMW98

SM 位	说　　明
SMW98	当扩展总线出现校验错误时，该处每次增加 1。当系统得电或用户程序写入零时，可以进行清零

19. SMB131～SMB165：HC3、HC4 和 HC5 寄存器

如表 B-18 所示，SMB131～SMB165 用于监视和控制高速计数器 HC3、HC4、HC5 的操作。

表 B-18　特殊存储器字节 SMB131～SMB165

SM 位	说　　明
SMB131～SMB135	保留
SM136.0～SM136.4	保留
SM136.5	HC3 当前计数方向状态位：1 = 增计数
SM136.6	HC3 当前值等于预设值状态位：1 = 等于
SM136.7	HC3 当前值大于预设值状态位：1 = 大于
SM137.0～SM137.2	保留
SM137.3	HC3 方向控制位：1 = 增计数
SM137.4	HC3 更新方向：1 = 更新方向
SM137.5	HC3 更新预设值：1 = 将新预设值写入 HC3 预设值
SM137.6	HC3 更新当前值：1 = 将新当前值写入 HC3 当前值
SM137.7	HC3 启用位：1 = 启用
SMD138	HC3 新初始值
SMD142	HC3 新预置值
SM146.0～SM146.4	保留
SM146.5	HC4 当前计数方向状态位：1 = 增计数
SM146.6	HC4 当前值等于预设值状态位：1 = 等于

（续）

SM 位	说　明
SM146.7	HC4 当前值大于预设值状态位：1 = 大于
SM147.0	复位的有效电平控制位：0 = 复位为高电平有效，1 = 复位为低电平有效
SM147.1	保留
SM147.2	正交计数器的计数速率选择：0 =4 × 计数速率；1 =1 × 计数速率
SM147.3	HC4 方向控制位：1 = 增计数
SM147.4	HC4 更新方向：1 = 更新方向
SM147.5	HC4 更新预设值：1 = 将新预设值写入 HC4 预设值
SM147.6	HC4 更新当前值：1 = 将新当前值写入 HC4 当前值
SM147.7	HC4 启用位：1 = 启用
SMD148	HC4 新初始值
SMD152	HC4 预置值
SM156.0 ~ SM156.4	保留
SM156.5	HC5 当前计数方向状态位：1 = 增计数
SM156.6	HC5 当前值等于预设值状态位：1 = 等于
SM156.7	HC5 当前值大于预设值状态位：1 = 大于
SM157.0 ~ SM157.2	保留
SM157.3	HC5 方向控制位：1 = 增计数
SM157.4	HC5 更新方向：1 = 更新方向
SM157.5	HC5 更新预设值：1 = 将新预设值写入 HC5 预设值
SM157.6	HC5 更新当前值：1 = 将新当前值写入 HC5 当前值
SM157.7	HC5 启用位：1 = 启用
SMD158	HC5 新初始值
SMD162	HC5 预置值

20. SMB166 ~ SMB185：PTO0、PTO1 包络定义表

如表 B-19 所示，SMB166 ~ SMB194 用来显示包络步的数量和包络表的地址以及 V 存储器区中表的地址。

表 B-19　特殊存储器字节 SMB166 ~ SMB185

SM 位	说　明
SMB166	PTO0 的包络步当前计数值
SMB167	保留
SMW168	PTO0 的包络表 V 存储器地址（从 V0 开始的偏移量）
SMB170	线性 PTO0 状态字节
SMB171	线性 PTO0 结果字节
SMD172	指定线性 PTO0 发生器工作在手动模式时产生的频率。频率是一个以 Hz 为单位的双整型值。SMB172 是 MSB，而 SMB175 是 LSB

（续）

SM 位	说　　明
SMB176	PTO1 的包络步当前计数值
SMB177	保留
SMW178	PTO1 的包络表 V 存储器地址（从 V0 开始的偏移量）
SMB180	线性 PTO1 状态字节
SMB181	线性 PTO1 结果字节
SMD182	指定线性 PTO1 发生器工作在手动模式时产生的频率。频率是一个以 Hz 为单位的双整型值。SMB182 是 MSB，而 SMB178 是 LSB

附录 C　S7-200 的错误代码

1. 致命错误代码和消息

严重错误将导致 S7-200 停止执行程序。依据错误的严重性，一个致命错误会导致 S7-200 无法执行某个或所有功能。处理致命错误的目标是使 S7-200 进入安全状态，S7-200 由此可以对存在的错误条件的相关询问做出响应。

当检测到致命错误时，S7-200 执行以下任务：

1）进入 STOP 模式；

2）点亮 SF/DIAG（红色）LED 指示灯和停止 LED 指示灯；

3）断开输出。

这种状态将会持续到错误清除之后。在主菜单中使用菜单命令“PLC”→“信息”可查看错误代码。表 C-1 列出了从 S7-200 上可读到的致命错误代码及其描述。

表 C-1　从 S7-200 读出的致命错误代码及消息

错误代码	说　　明
0000	无致命错误
0001	用户程序校验和错误
0002	编译后的梯形图程序校验和错误
0003	扫描看门狗超时错误
0004	永久存储器失效
0005	永久存储器上用户程序校验和错误
0006	永久存储器上组态参数（SDB0）校验和错误
0007	永久存储器上强制数据校验和错误
0008	永久存储器上默认输出表值校验和错误
0009	永久存储器上用户数据 DB1 校验和错误
000A	存储器卡失灵
000B	存储器卡上用户程序校验和错误
000C	存储卡组态参数（SDB0）校验和错误

（续）

错误代码	说　明
000D	存储器卡强制数据校验和错误
000E	存储器卡默认输出表值校验和错误
000F	存储器卡用户数据 DB1 校验和错误
0010	内部软件错误
0011①	比较触点间接寻址错误
0012①	比较触点浮点值错误
0013①	程序不能被该 S7-200 理解
0014①	比较触点范围错误

① 表示比较触点错误是唯一的一种既能产生致命错误又能产生非致命错误的错误。产生非致命错误是因为存储错误的程序地址。

2. 运行程序错误

在程序的正常运行中，可能会产生非致命错误（如寻址错误）。在这种情况下，CPU 产生一个非致命运行时刻错误代码。表 C-2 列出了这些非致命错误代码及其描述。

表 C-2　运行程序错误

错误代码	说　明
0000	无致命错误；无错误
0001	在执行 HDEF 指令之前，已经执行 HSC 指令
0002	输入中断分配冲突，已分配给 HC
0003	到 HC 的输入分配冲突，已分配给输入中断或其他 HC
0004	试图执行在中断子程序中不允许的指令
0005	尝试在完成第一个 HSC/PLS 之前执行第二个带有相同号码的 HSC/PLS，与主程序中的 HSC/PLS 发生中断例行程序冲突
0006	间接寻址错误
0007	TODW（写实时时钟）或 TODR（读实时时钟）数据错误
0008	用户子程序嵌套层数超过规定
0009	在程序执行 XMT 或 RCV 时，端口 0 又执行另一条 XMT/RCV 指令
000A	在同一 HC 执行时，又企图用 HDEF 指令再定义该 HC
000B	在端口 1 上同时执行数条 XMT/RCV 指令
000C	时钟存储卡不存在
000D	试图重新定义正在使用的脉冲输出
000E	PTO 包络段数设为 0
000F	比较触点指令中的非法数字值
0010	在当前 PTO 操作模式中，命令未允许
0011	非法 PTO 命令代码
0012	非法 PTO 包络表

（续）

错误代码	说　明
0013	非法 PID 回路参数表
0091	范围错误（带地址信息）：检查操作数范围
0092	指令的计数域出错（带计数信息）：确认最大计数大小
0094	范围错误（带地址信息）：写无效存储器
009A	用户中断程序试图转换成自由端口模式
009B	非法指针（字符串操作中起始位置值指定为 0）
009F	无存储卡或存储卡无响应

3. 编译规则错误

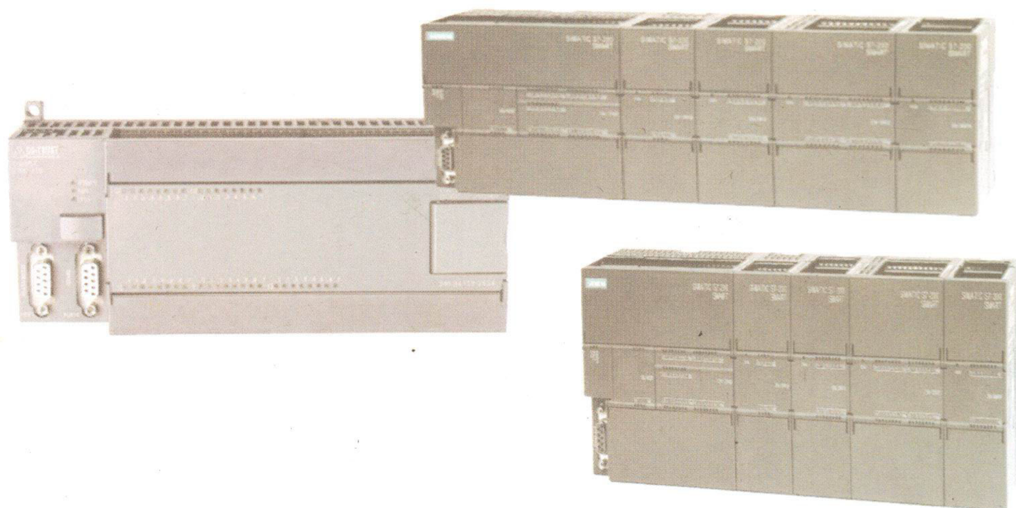
下载一个程序时，CPU 将编译该程序。如果 CPU 发现程序违反编译规则（如非法指令），CPU 就会停止下载程序，并生成一个非致命编译规则错误代码。表 C-3 列出了违反编译规则所生成的这些错误代码及其描述。

表 C-3　编译规则错误

错误代码	说　明
0080	程序太大，无法编译；减少程序大小
0081	堆栈溢出；将程序分成多个程序段
0082	非法指令；检查指令助记符
0083	主程序中缺失 MEND 或存在不允许的指令：添加 MEND 指令或删除错误指令
0084	保留
0085	缺失 FOR；添加 FOR 指令或删除 NEXT 指令
0086	缺失 NEXT；添加 NEXT 指令或删除 FOR 指令
0087	缺失标签（LBL、INT、SBR）；添加相应的标签
0088	子程序中缺失 RET 或存在不允许的指令：将 RET 添加到子程序末尾或删除错误指令
0089	中断例行程序中缺失 RETI 或存在不允许的指令：将 RETI 添加到中断例行程序末尾或删除错误指令
008A	保留
008B	从/向一个 SCR 段的非法跳转
008C	重复标签（LBL、INT、SBR）；重命名其中一个标签
008D	非法标签（LBL、INT、SBR）；确保不超出所允许的标签数目
0090	非法参数；确认指令允许的参数
0091	范围错误（带地址信息）；检查操作数范围
0092	指令的计数域出错（带计数信息）；确认最大计数大小
0093	FOR/NEXT 嵌套层数超出范围
0095	无 LSCR 指令（装载 SCR）
0096	无 SCRE 指令（SCR 结束）或 SCRE 前面有不允许的指令
0097	用户程序包含非数字编码的和数字编码的 EV/ED 指令

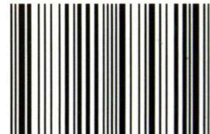
（续）

错误代码	说　明
0098	在运行模式进行非法编辑（试图编辑非数字编码的 EV/ED 指令）
0099	隐含程序段太多（HIDE 指令）
009B	非法指针（字符串操作中起始位置值指定为 0）
009C	超出最大指令长度
009D	SDB0 中检测到非法参数
009E	PCALL 字符串太多
009F～00FF	保留



地址：北京市百万庄大街22号
邮政编码：100037
电话服务
社服务中心：010-88361066
销售一部：010-68326294
销售二部：010-88379649
读者购书热线：010-88379203
网络服务
教材网：http://www.cmpedu.com
机工官网：http://www.cmpbook.com
机工官博：http://weibo.com/cmp1952
封面无防伪标均为盗版

上架指导 工业技术／自动化技术／PLC
ISBN 978-7-111-47481-4
策划编辑◎徐明煜



定价：49.90元